ARI Contractor Report 2005-09

An Intelligent Tutoring System Approach to Adaptive Instructional Systems

James Ong and Sowmya Ramachandran
Stottler Henke Associates, Inc.

This report is published to meet legal and contractual requirements and may not meet ARI's scientific or professional standards for publication.

September 2005

United States Army Research Institute for the Behavior and Social Sciences

REPORT DOCUMENTATION PAGE

1. REPORT DATE (dd-mm-yy) September 2005	2. REPORT TYPE Final	3. DATES COVERED (from. . . to) March 2000 – October 2000

4. TITLE AND SUBTITLE An Intelligent Tutoring System Approach to Adaptive Instructional Systems	5a. CONTRACT OR GRANT NUMBER DASW01-00-C-3015 5b. PROGRAM ELEMENT NUMBER 622785
6. AUTHOR(S) James Ong and Sowmya Ramachandran (Stottler Henke Associates, Inc.)	5c. PROJECT NUMBER A790 5d. TASK NUMBER 206 5e. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Stottler Henke Associates, Inc. 1660 South Amphlett Blvd., Suite 350 San Mateo, CA 94402	8. PERFORMING ORGANIZATION REPORT NUMBER Report No 204: SWIFT Option Final Report
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U. S. Army Research Institute for the Behavioral & Social Sciences ATTN Rotary Wing Aviation Research Unit 2511 Jefferson Davis Highway Arlington, VA 22202	10. MONITOR ACRONYM ARI 11. MONITOR REPORT NUMBER Contractor Report 2005-09

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES Contracting Officer's Representative and Subject Matter POC: William R. Howse. This report is published to meet legal and contractual requirements and may not meet ARI's scientific and/or professional standards for publication.

14. ABSTRACT *(Maximum 200 words)*:
Report developed under a Small Business Innovation Research Program 99.2 contract for topic OSD-99-04. Training programs provide students with deliberately selected learning experiences, so they can acquire and retain knowledge and skills. Intelligent Tutoring Systems (ITSs) are computer-based training systems that mimic human instructors to provide automated, one-on-one instruction. Most ITSs developed so far have applied a limited set of strategies for adapting instruction to differences among individual students. During this Phase I SBIR project, we developed a generic model of adaptive instructional systems that is designed to be broadly applicable across a wide range of training domains. We then applied this generic model as a framework for describing how AIS capabilities could be added to the Intelligent Flight Trainer (IFT), a helicopter training simulator deployed at Ft. Rucker, Alabama. Finally, we developed a limited, proof-of-concept software prototype to illustrate elements of this model. The generic AIS model appears plausible, feasible, and useful, and the software prototype provides additional encouragement.

15. SUBJECT TERMS
SBIR report, adaptive instruction, intelligent tutoring systems, helicopter pilot training, command and control

SECURITY CLASSIFICATION OF			19. LIMITATION OF ABSTRACT	20. NUMBER OF PAGES	21. RESPONSIBLE PERSON
16. REPORT Unclassified	17. ABSTRACT Unclassified	18. THIS PAGE Unclassified	Unlimited	54	Ellen Kinzer Technical Publication Specialist 703-602-8047

FOREWORD

Training programs provide students with deliberately selected learning experiences, so they can acquire and retain knowledge and skills. Intelligent Tutoring Systems (ITSs) are computer-based training systems that mimic human instructors to provide automated, one-on-one instruction. Although ITSs typically adapt their instruction in response to individual student differences *to some degree*, most ITSs developed so far have applied a limited set of strategies for doing so, focusing primarily on *microadaptive* algorithms that consider the student's solution history.

However, the most effective instructional method for one group of students may not be the best for other types. Thus, the effectiveness of instructional systems can be improved by incorporating algorithms that adapt instruction to individual differences. Although much of the research in learning and individual differences so far has focused on interactions between student aptitude level and learning environment, many other student attributes can also be usefully considered when making instructional decisions. These other attributes include cognitive styles, personality types, mental and emotional state, student experiences, and learning style. Consideration of these other attributes, however, requires the development of *adaptive instructional systems* (AIS) models that specify the data and algorithms required to *assess* (or estimate) these student attributes and *apply* these estimates to make better instructional decisions.

During this Phase I SBIR project, we developed a generic model of adaptive instructional systems that is designed to be broadly applicable across a wide range of training domains. We then applied this generic model as a framework for describing how AIS capabilities could be added to the Intelligent Flight Trainer (IFT), a helicopter training simulator deployed at Ft. Rucker, Alabama. Finally, we developed a limited, proof-of-concept software prototype to illustrate some elements of this model. This project provides a concrete test case for exploring the utility and feasibility of implementing AIS capabilities, and it provides a platform for exploring the AIS design issues in other training domains. We propose to continue AIS research in command and control domains, by enhancing existing tutoring systems developed by SHAI to incorporate AIS capabilities.

AN INTELLIGENT TUTORING SYSTEM APPROACH TO ADAPTIVE INSTRUCTIONAL SYSTEMS

EXECUTIVE SUMMARY

Research Requirement:

This research is motivated by the importance of incorporating sophisticated adaptive strategies within intelligent tutoring systems to improve their effectiveness. By *adaptive instruction systems (AIS)*, we refer to intelligent tutoring systems that select and present diagnostic and learning experiences that are tailored to each student's individual characteristics, to support more effective learning. Generic and domain-specific models of adaptive instructional systems can:

- Help researchers identify important research issues that relate to adaptive instruction. For example, the AIS model may suppose relationships among observable data, learning and cognitive styles, personality types, mental and emotional state, instructional strategies, and/or student performance that deserve experimental validation.
- Provide guidance to researchers, knowledge engineers, and software designers when designing and implementing AIS models and software systems for other training domains. For example, these models can help identify potential adaptive instructional strategies, observable data, and student model attributes that should be incorporated within specific AISs.
- Encourage the development of AIS software modules and knowledge bases that can be re-used across different, domain-specific AISs. We believe that many AIS strategies are broadly applicable across training domains, so it may be possible to create re-usable AIS software modules that implement these generic strategies.

Procedure:

During this six month phase I SBIR project, Stottler Henke Associates, Inc. (SHAI):

- Reviewed relevant psychology and intelligent tutoring systems research literature,
- Reviewed U.S. Army training documents and lay press books on helicopter piloting to become familiar with the training domain and current training procedures,
- Visited Ft. Rucker Army base on April 14 and May 16, 2000 and interviewed helicopter instructor pilots (IPs) to become familiar with training practices.
- Designed the generic AIS model,
- Identified ways in which AIS capabilities could be applied to helicopter pilot training, and
- Developed a limited software prototype.

During the phase I option period, Stottler Henke Associates, Inc. (SHAI) began the Phase II effort by:

- Reviewing MAMIDS literature,
- Reviewed research literature on personality and learning styles,
- Determined how that work could be incorporated into the Phase II System, and
- Determined how the C4I ITS could benefit from increased adaptive capabilities.

Findings:

Adaptive Instructional Systems (AIS) capabilities show promise as an effective method of improving automated instruction, by adapting instructional decisions to individual differences among students. The generic AIS model appears plausible, feasible, and useful, and the software prototype provides additional encouragement. In addition, some of the AIS knowledge structures and algorithms may be generic across different, domain-specific AIS systems, making it possible to re-use some software and data objects across AIS systems.

Utilization and Dissemination of Findings:

We propose to use the AIS model developed during this project as the basis for further research in AIS capabilities in other domains, such as command and control. Specifically, we propose to continue this research using Phase II SBIR funding to add adaptive instructional capabilities to existing tutoring systems developed by SHAI for teaching tactical decision-making and command and control. These systems include the Tactical Action Officer (TAO) ITS developed by SHAI for the U.S. Navy, and the C4I ITS under development by SHAI for STRICOM that interfaces with the FBCB2 command and control system.

TABLE OF CONTENTS

1. Research Objective

Training programs provide students with deliberately selected learning experiences, so they can acquire and retain knowledge and skills. These experiences include *natural feedback* provided by training simulations as well as presentations, explanations, hints, feedback, and other interventions selected and presented by the instructor. Intelligent Tutoring Systems (ITSs) are computer-based training systems that mimic human instructors to provide automated, one-on-one instruction. Although ITSs typically adapt their instruction in response to individual student differences *to some degree*, most ITSs developed so far have applied a limited set of strategies for doing so, focusing primarily on *microadaptive* algorithms that consider the student's solution history.

However, the most effective instructional method for one group of students may not be the best for other types, and the effectiveness of instructional systems can be improved by incorporating algorithms that adapt instruction to individual differences. Much of the research in automated instruction and individual differences so far has focused on interactions between student aptitude level and learning environment. However, many other student attributes can also be usefully considered when making instructional decisions, such as cognitive styles, personality types, mental and emotional state, student experiences, and learning style. Consideration of these other attributes, however, requires the development of *adaptive instructional systems* (AIS) models that specify the data and algorithms required to *assess* (or estimate) these diverse student attributes and *apply* these estimates to make better instructional decisions. The development of generic and domain-specific models of adaptive instructional systems can:

- Help researchers identify important research issues that relate to adaptive instruction. For example, the AIS model may suppose relationships among observable data, learning and instructional strategies, and/or student performance that deserve experimental validation.

- Provide guidance to researchers, knowledge engineers, and software designers when designing and implementing AIS models and software systems for a wide range of training domains. For example, these models can help identify potential adaptive instructional strategies, observable data, and student model attributes that should be incorporated within specific AISs.

- Encourage the development of AIS software modules and knowledge bases that can be re-used across different, domain-specific AISs. We believe that many AIS strategies are broadly applicable across training domains, so it may be possible to create re-usable AIS software modules that implement these generic strategies.

2. Project Summary

During this Phase I SBIR project, Stottler Henke Associates, Inc. (SHAI) developed a generic model of adaptive instructional systems that is designed to be broadly applicable across a wide range of training domains. We then applied this generic model as a framework for describing how AIS capabilities could be added to the Intelligent Flight Trainer (IFT), a helicopter training simulator deployed at Ft. Rucker, Alabama. Finally, we developed a limited, proof-of-concept prototype to illustrate elements of this model. This project provided a concrete test case for exploring the utility and feasibility of implementing AIS capabilities, and it provides a platform for exploring the AIS design issues in other training domains. We propose to continue AIS research in command and control domains, by enhancing existing tutoring systems developed by SHAI to incorporate AIS capabilities.

During this project, SHAI performed the following tasks:

- Reviewed relevant psychology and intelligent tutoring systems research literature,
- Reviewed U.S. Army training documents and lay press books on helicopter piloting to become familiar with the training domain and current training procedures,
- Visited Ft. Rucker Army base on April 14 and May 16, 2000, and interviewed helicopter instructor pilots (IPs) to become familiar with training practices,
- Designed the generic AIS model,
- Identified ways in which AIS capabilities could be applied to helicopter pilot training, and
- Developed a limited software prototype that illustrated some elements of the AIS model.

3. AIS Model

3.1 AIS Software Architecture

Figure 1 shows our generic AIS software architecture.

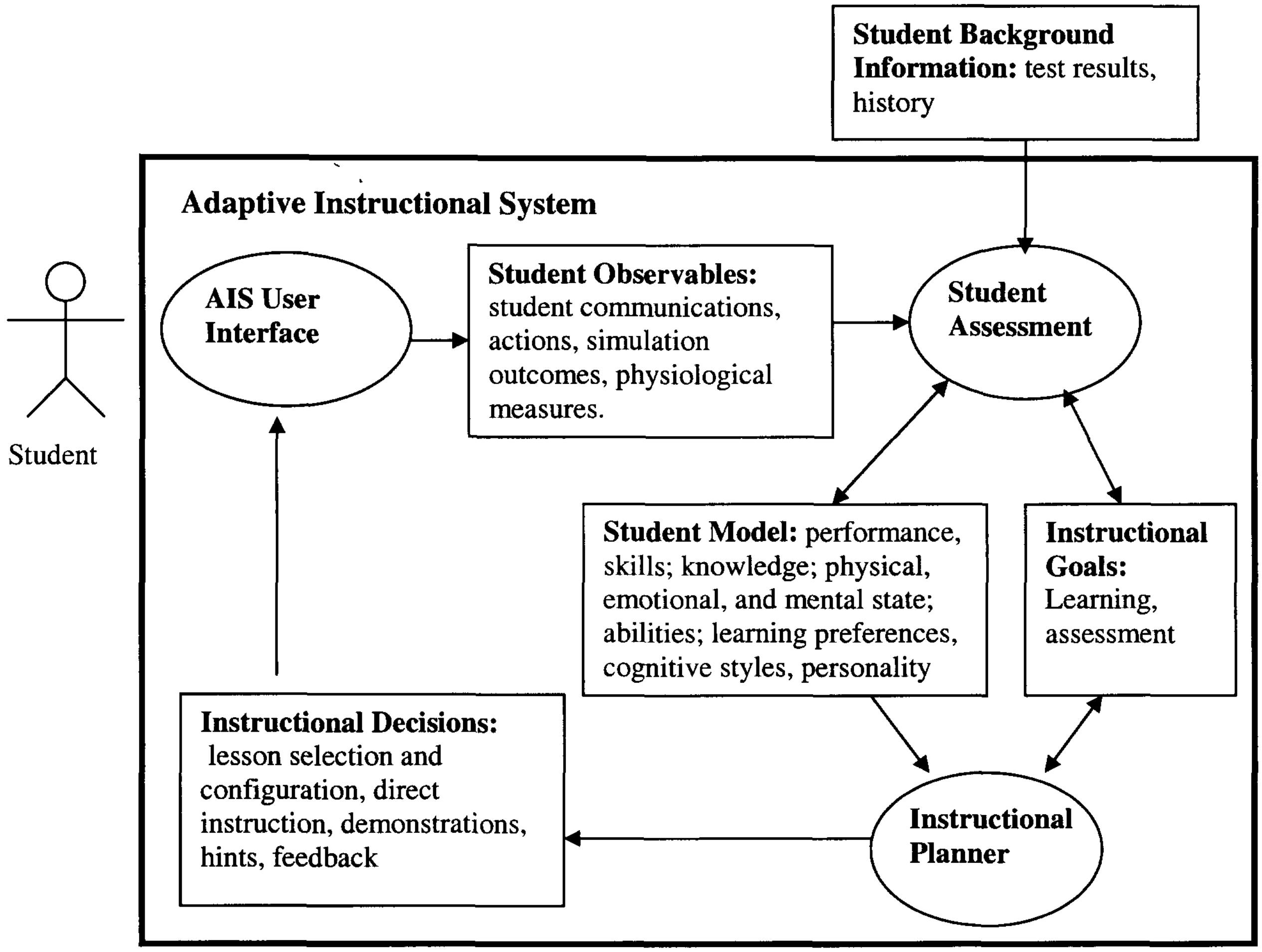

Figure 1 - AIS software architecture. (arrows show inferential and decision-making relationships)

AIS is comprised of the following software modules (shown in Figure 1 as ovals) and data objects (shown as rectangles):

- **Simulator / AIS User Interface** - This software module includes the hardware/software simulator or interactive problem-solving environment with which the student interacts. It also includes the user interface by which the tutoring component of the AIS and the student exchange information.
- **Instructional Decisions** - These decisions include the selection of Simulator parameters or scenarios presented by the Simulator. These decisions also include selection of the content and format of instructional interventions such as hints, feedback, direction instruction, and questions presented through the AIS User Interface.

- **Instructional Planner** - This software module applies adaptive instructional strategies to make instructional decisions based on its student model that includes assessments of the student's performance and estimates of the student's skills, knowledge, personality, cognitive abilities and styles, learning preferences, physical, mental, and emotional state.
- **Student Model** - Ideally, the Instructor Model would base its decisions on the "actual" attributes of the student. However, these attributes represent internal states of the student that can only be estimated. The Student Model represents these estimates, as well as objective evaluations of the student's performance.
- **Instructional Goals** - The AIS pursues two primary goals: providing the student with effective learning experiences and assessing the student's skills, knowledge, and other student model attributes to provide a valid basis on which to select learning experiences. To prioritize and pursue these goals, the AIS reasons using explicit knowledge representations of goals, subgoals, and plans which achieve each type of goal.
- **Student Assessment** - This software module maintains the Student Model by estimating student attributes from observable data collected by the AIS and background information about the student. This assessment can also be biased by current estimates of other student attributes stored in the Student Model.
- **Student Observables** - This data includes information communicated by the student to the AIS, actions carried by the student within the Simulator, simulation outcomes caused by student actions, and physiological measures.

The AIS can be described as an intelligent agent because it combines the ability to assess its environment (i.e., assess student attributes from observables) and carry out actions on the environment (i.e., make instructional decisions).

This architecture is generic and would be elaborated for each domain-specific AIS. To develop a domain-specific AIS, the designer must populate the software modules and data models with algorithms and knowledge representations that achieve the system's training objectives within schedule and budget constraints. Although some of the algorithms and data objects to be included in an AIS will be specific to each specific training domain, many algorithms and data objects will be generally applicable to a broad set of training domains, as described in the remainder of section 3.

3.2 Student Model

The AIS designer must select student model attributes that can be:

- *Estimated* by the Student Assessment Module from Student Observables and Student Background Information, and
- *Applied effectively* by the Instructor Model to adapt Instructional Decisions to individual differences.

This section describes generally useful student model components that can be estimated and applied effectively to control instructional decision-making.

3.2.1 Knowledge

Knowledge is a prerequisite for any skill: before a task can be performed, the subject must possess the knowledge needed to understand the relevant facts, procedures, and cause-and-effect relationships. However, practice is typically required to improve the speed, accuracy, and automaticity with which this knowledge can be recalled and applied to solve problems and perform tasks. For example, Anderson's ACT* model of learning (1983) posits three stages of learning: declarative knowledge, procedural skills, and automatic skills.

A number of formalisms have been advanced for representing this knowledge, including production rule-based systems (Anderson, 1983) and schema-based systems that reason using specific episodes (cases) or prototypes (scripts) (Schank, 1977). Many software systems have been developed that employ these representations.

Many instructional theories and systems presume that the instructor or instructional system possesses a model of the knowledge structures required by the student, and that the goal of the instruction is to *transmit* these knowledge structures to the student. These approaches are most appropriate when the task is relatively procedural, such as Algebra, so that well-accepted facts and procedures can be specified for the domain. By contrast, *constructivists* believe that learning experiences should enable subjects to *construct* their own individual knowledge structures that are compatible with each subject's unique experiences and the existing knowledge structures (Jonassen, 1999). Thus, the goal of constructivist instruction is to help the student think about the experience to facilitate knowledge construction. For example, the instructor could make observations and pose questions to stimulate the student's thinking. Constructivist approaches are appropriate in complex domains in which it is difficult to circumscribe the knowledge needed to perform a task into well-isolated set of facts and procedures.

3.2.2 Skills

In our AIS model, a *skill* is:

- the ability to carry out a task,
- that can be acquired through training, and
- requires the application of a combination of perceptual, cognitive, and motor processes.

The level of each skill for each student can be characterized by its:

- *proficiency* - the subject's ability to attain high levels of task performance, as measured by speed, accuracy, etc.
- *automaticity* - the subject's ability to perform a task "automatically", in parallel with other tasks with little apparent effort or allocation of attention.

Tasks are often composed of simpler subtasks that must be mastered and applied sequentially and/or in parallel. Parallel execution of these tasks may require overlearning of the subtasks so that each subtask can be performed proficiently with few attentional resources required.

At SHAI, we have developed practical, intelligent tutoring systems that employ relatively shallow representations of knowledge and skills. These representations support case-based instructional systems in which it is unnecessary to build an expert system that automatically finds solutions to problems in the domain. Instead, case-based instructional systems require only enough knowledge of the domain to support reasoning about instructional decisions. This knowledge may include skill-subskill relationships (e.g., skill S_1 requires subskills S_2, S_3, and S) and pre-requisite knowledge relationships (e.g., skill S_1 has pre-requisite knowledge K_1, K_2, and K_3.)

3.2.3 Task Performance

A student's performance describes how well he or she performed a task. There are several types of criteria on which these assessments can be based.

The traditional measures of task performance are speed and accuracy. For tasks where the desired result of performing the task may be complex, it may be useful to view performance as the degree to which the student *achieved the desired result or goal state.* In a sense, this can be thought of as a type of accuracy measurement. For example, the performance of a student helicopter pilot carrying out a specific maneuver can be quantified by the difference between actual and target values of the flight variables over time such as altitude, heading, and x-y position.

Although this type of performance assessment is frequently straight-forward to implement, it does not always provide sufficiently detailed information needed to identify the student model attributes such as subskills or pre-requisite knowledge in which the student is weak. This detailed information is needed to focus instruction on the underlying causes of the weak task performance. Acquiring this detailed information may require the (human or automated) instructor to carry out diagnostic actions and/or estimate these attributes using other observable data and student model estimates.

Performance assessment can also be based on the degree to which the student's actions or decision-making *conforms to the "correct solution."* Use of this criteria presumes that there is just one (or at most a small number) of correct solutions with which the student's actions can be compared. This is not true of helicopter piloting because different combinations of flight control actions can achieve the desired outcome. However, in many domains it may be plausible to assume that there is a single correct solution to a given problem, at least in a given scenario. This criteria also assumes that the significant student actions can be observed. However, in many domains, the subject must carry out many (unobservable) perceptual or cognitive processes prior to carrying out each observable action. In these domains, assessment of the observable actions may provide only partial insight into whether the subject is carrying out the correct sequence of actions.

Performance assessment can also include *characteristic error patterns* observed in the student's actions. For example, an instructor may observe that a student helicopter pilot tends to overcontrol the cyclic when carrying out a maneuver. By augmenting its overall assessment of the student's task performance with specific error patterns, the instructional system can identify knowledge or skill deficiencies with more precision, and therefore select highly targeted instructional interventions to address the student's performance problems.

3.2.4 Physical, Emotional, and Mental State

A subject's task performance level is correlated with his or her skill proficiency, but it is also affected by the subject's current mental or physical state. For example, a subject will perform more poorly if he or she is mentally or physically fatigued, task-overloaded, or unable to devote his or her full attention to the task. Emotional state can also affect performance. For example, a student who is stressed or anxious may fixate on certain stimuli and become less able to respond to other stimuli.

3.2.5 Abilities and Mediators

Considerable research in psychology has been devoted to searching for relationships between general (task-independent) abilities and task-specific performance or skill-acquisition. However, there are many theories about the number and type of these general abilities. Researchers disagree as to whether there is a single general intelligence (Herrnstein, 1994); two intelligences: fluid and crystallized (Catell, 1971); three (Eysenck, 1986); or many (Gardner, 1983).

We do not believe that it is necessary to resolve this general issue in order to design effective adaptive instructional systems. Instead, we take a pragmatic approach: the set of abilities to be modeled in an AIS should depend only upon whether individual differences in those abilities can be estimated and usefully incorporated within the AIS to improve instructional decision-making.

Therefore, the set of abilities modeled might include both broad and highly specific characteristics and may vary across AISs designed for different training domains. Broad characteristics include:

- Basic cognitive abilities: perceptual, situation assessment, decision-making, motor control
- Cognitive mediators: attention, short-term memory, long-term memory
- Reasoning methods: pattern recognition, inference, recall
- Associative learning (AL) skills: verbal, quantative, spatial (Anderson, 1983)

3.2.6 Learning Preferences, Cognitive Styles, and Personality

The subject's response to different training methods and learning experiences, and therefore his or her skill and knowledge acquisition, is affected by the subject's personality, cognitive style, and learning preferences.

Examples of learning preferences include:

- **Part-task vs. whole-task training** - Complex tasks require the combined execution of simpler sub-tasks. Some students may learn faster by first practicing each subskill individually and then integrating the subskills later on, whereas other student's may learn faster by practicing the entire task from the beginning. In general, there may be more than one way to decompose a task into subtasks. For example, hovering a helicopter can be decomposed into the subtasks of controlling the cyclic, the collective, and the pedals. An advantage of this task decomposition is that it is possible for an instructor (or automated training system) to assume responsibility for one or more of the controls in order to simplify the task. The hover task can also be decomposed into a different set of subtasks: perceiving the helicopter's altitude, controlling the altitude, perceiving the helicopter's xy position, controlling the position, etc. This decomposition has the advantage that different subskills may underly each of the subtasks, so identifying the problematic subtasks helps to identify problematic subskills.
- **Self-direction vs. programmed instruction** - Some students learn more quickly when they can control the learning experiences provided by the training system. Other students

learn more effectively when the program selects the student's learning experiences. Cronback and Snow (1977) report that high aptitude students tend to benefit from the freedom to explore on their own, whereas low-ability students fare better when the instructional decisions are made for them.

- **Rule application vs. rule induction** - Training systems can vary in how they present feedback to the student. Systems can communicate to the student the relevant rule and its application to a specific problem (rule application), or they can communicate hints from which the student must infer the rule application. (Shute, 1992) describes an experiment that showed interactions among the learning environment (rule application vs. rule induction), student associative learning skills (low vs. high), and task type (declarative vs. procedural vs. generative design) when predicting performance.

Examples of cognitive styles include:

- **Dominant thinking styles** (Masie, 1997) – These can be classified into the following four categories: 1. Reflective, 2. Conceptual, 3. Practical, and 4. Creative. Although people have dominant thinking styles, they also use different styles at different times. These thinking styles have an implication to training. For instance, people who are predominantly conceptual in their thinking like to understand the whole picture and respond better to structured instruction. This implies that they should be presented with overviews before focusing on particular aspects of the domain. On the other hand, practical thinkers, who like to focus on useful, practical information, should be presented with clear learning objectives, and allowed to jump straight to hands-on exercises.

- **Case-based vs. model-driving reasoning** - Some students reason predominantly by remembering concrete episodes (or cases), whereas other subjects reason using abstract principles, models, or rules. Although few subjects use one reasoning method to the exclusion of the other, many subjects will prefer one method over the other and will respond more positively to instruction delivered in the preferred form.

A subject's personality can affect his or her perceptual and cognitive processes, thereby affecting skill acquisition and performance. For example:

- Personality attributes can skew the subject's **situation assessment** and **decision-making**. For example, an overly anxious subject may over-react to perceived threats and under-react to possible opportunities. A cautious learner may adhere to well-practiced but sub-optimal behavioral patterns, rather than experiment with new behaviors that may produce better results. These relationships affect the subject's ability to acquire certain skills and knowledge and could be considered by an instructional system when selecting lessons for the student or when estimating the student's current mastery of knowledge and skills.

Methodologies for creating models, which relate these factors to performance, however, have only begun to emerge. For example, Hudlicka (1999) describes a Methodology for Analysis and Modeling of Individual Differences (MAMID). MAMID represents a generic approach for representing cognitive, personality, and affective factors in human performance models. We are optimistic that this methodology can help create models that relate cognitive, affective, and personality factors to cognitive processes that affect task performance.

3.3 Student Assessment

Observables, such as test results, student actions, performance, communications, and physiological measurements are causally related (directly and indirectly) to attributes of the student such as his or her abilities, skills, knowledge, and mental state. In general, the student attributes which are the most reliable determinants of effective, adaptive instruction, are also those that cannot be measured directly but must be estimated by the Student Assessment module. Figure 3 shows the causal relationships between observables (in bold boxes) and data that can only be estimated.

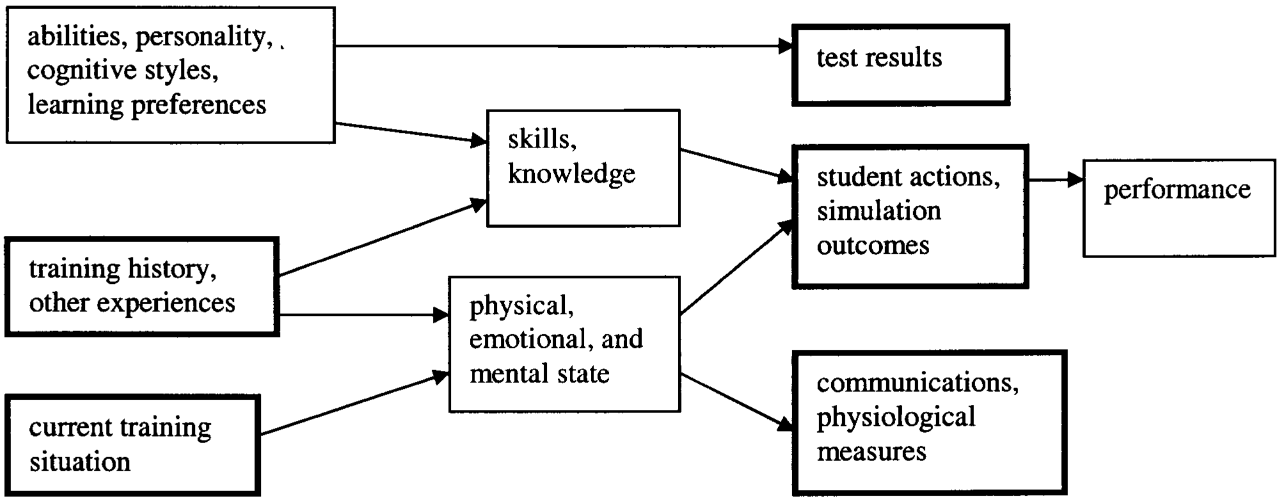

Figure 2 - Causal relationships among student attributes and observables

Because the relationship between observables and student attributes is many-to-many, it is usually not possible to simply deduce the value of a student attribute from the value of an observable measurement. For example, poor task performance may be caused by low skill proficiency relevant to the task, but it may also be caused by fatigue or low motivation. Also, successful performance of a task may require a combination of subskills, so an instructional system must also determine which specific subskills are weak. Identifying the cause of symptoms when many-to-many relationships exist, is a classic artificial intelligence problem which has been tackled using techniques such as rule-based programming, case-based reasoning, and Bayesian inference.

3.3.1 Assessing Performance

Performance can be assessed by evaluating the actions performed by the student or evaluating the states (or conditions) achieved by the student. For example, a command and control training system might evaluate the student based on the outcome of the engagement or on the student's actions.

These actions can be discrete (e.g., flipping a switch, issuing an order) or they can vary continuously over time (e.g., the position of the helicopter collective vs. time). State changes include those that, by definition, result from an action (e.g., flipping a switch causes the switch to be in the ON position) as well as states that are achieved by the action (e.g., flipping the heater switch causes the temperature of the heater to rise only if power is supplied to the heater).

Performance assessment based upon continuously-varying actions and/or states requires analysis of time-series data to identify significant time-series data patterns to estimate proficiency and detect error modes. This time-series analysis may be combined with symbolic pattern matching to reason about data patterns found in the time-series. For example, the "over-controlling" of the cyclic can be detected by detecting symptomatic patterns in the helicopter's trajectory.

Most intelligent tutoring systems have been developed for training domains where the actions are discrete. A number of methods have been developed for these types of domains, including:

- **Model tracing** - The system encodes a cognitive model, encoded as a set of production rules that is capable of solving problems in the way students are to solve the problems. This model is then used as the basis for evaluating student performance, and the system strives to explain incorrect solutions submitted by the student as perturbations on this model (e.g., missing or buggy rules). Specifically, the system tries to identify the set of rules (including some rules that may be incorrect) that the student might have applied that would yield the incorrect answer. Pursuing this approach is challenging (and usually expensive) because it requires the development of a cognitively plausible expert system that solves problems in the domain. This approach therefore is limited to relatively closed domains in which it is feasible to develop such an expert system. In addition, unambiguous interpretation of student actions is difficult because more than one sequence of production rules can produce a particular surface behavior that matches that of the student's.
- **Case-based reasoning** - SHAI has pursued case-based (or scenario-based) approaches to automated instruction where an instructor specifies a solution (or range of solutions) that are appropriate to each scenario presented to the student. Thus, the system is not necessarily capable of automatically generating solutions to problems in the domain (as in the model-tracing approach.) Instead, each scenario encodes one or more pattern-recognizers that detect sequences of actions and states that indicate the mastery (or lack of mastery) of certain knowledge or skills. ITS authoring tools enable instructors or subject matter experts to specify these pattern recognizers easily via graphical user interface.

3.3.2 Estimating Knowledge and Skills

Fine-grained assessment of knowledge and skills enables the training system to select highly specific instructional interventions that focus on the skills in need of improvement or the knowledge that requires reinforcement. For example, a skills assessment like "Student Bob is a poor helicopter pilot" is so broad that it provides little guidance for making instructional decisions. An assessment like "Student Bob has trouble with hovering" provides at least a starting point for focusing the instructor's attention. However, the instructor would still need to identify the specific reasons why the student's hovering skills are weak in order to determine the most appropriate instructional interventions.

In general, fine-grained estimates of student skills require more detailed assessments of student performance. For example, if the assessment of the student's performance hovering the helicopter simply produces a pass/fail rating, it is difficult to infer much about the student's skills, except that his or her hovering skill is high or low. However, if the performance assessment can determine error modes (e.g., overcontrolling, undercontrolling) or deficiencies in specific parameters of the task (e.g., maintaining altitude), the instructional system can identify candidate subskills that may be deficient.

Fine-grained knowledge and skills assessment cannot always be performed by assessing the performance level of a task that applies only the specific skill, for several reasons. First, there might not be a measurable task that corresponds to the skill. For example, one of the skills required to hover a helicopter is to maintain awareness of the height of the aircraft off the ground, requiring perceptual skills, and an ability to estimate the aircraft's distance off the ground from these visual cues. However, it may be difficult to assess these perceptual and distance estimation skills in isolation. In addition, it is inefficient to teach and estimate very fine-grained skills one-at-a-time.

To support more efficient learning, the AIS must be able to reason about skills and subskills. If the student's proficiency in S_1 is weak, the reason may be that the student has not become sufficiently proficient in one or more of the skill's subskills. Or, it may be that the student has not overlearned the subskills sufficiently to apply them in combination without becoming task-overloaded. Thus, the student must increase the automaticity with which he or she can perform the subskills. This reasoning leads to the following generic assessment strategies:

```
If the proficiency in skill S1 is low,
Then a possible cause may be that the proficiency of one of its
subskills is low.

If the proficiency in skill S1 is low,
Then a possible cause may be that the automaticity of one of its
subskills is low.
```

Figure 3 illustrates this strategy graphically. To determine whether a skill has become overlearned (automaticity is high), measure the task's performance when only partial attention can be devoted to the task, as shown by the vertical gray bar. If the task has been overlearned, the student's performance will approximate the benchmark performance level when full attention can be paid. If the task has been learned but not overlearned, the partial-attention performance level will be significantly below the benchmark performance level.

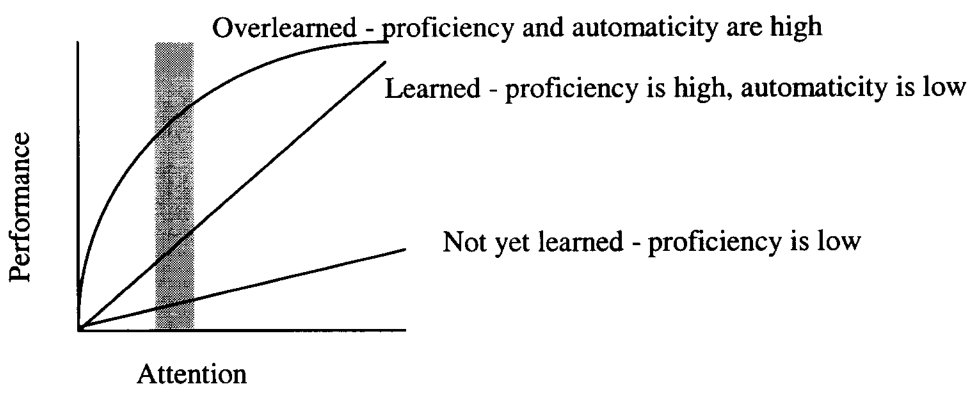

Figure 3 - Performance vs. attention by skill proficiency and automaticity

If a subject's proficiency in a skill is high, he or she can attain high performance when devoting attention to tasks that require the skill. However, in order to estimate the automaticity of a skill, the AIS can assess task performance when the task is carried out in parallel with another task that requires attentional resources. This yields the generic assessment strategy:

> `To estimate the automaticity of a skill, measure the difference between the performance of the skill in isolation and the performance of the skill in combination with another task.`

3.3.3 Estimating Physical, Emotional, and Mental State

With current technology, it is infeasible (or at least cost-prohibitive) for an AIS to rely upon vision or speech recognition to estimate the subject's physical, emotional, or mental state as a person would. However, an AIS can estimate these attributes by combining evidence provided by:

- physiological measures,
- the system's knowledge of the student's current situation and recent history (e.g., the student is likely to be fatigued because he or she has been performing difficult hovering maneuvers for the past two hours),
- the system's estimate of the subject's slowly-varying attributes, such as personality, and mental and physical ability, and
- patterns detected in the subject's actions that indicate (or suggest) certain states, especially when those actions are compared with typical patterns of action carried out by the subject.

3.3.4 Estimating Abilities and Mediators

Test batteries administered before or during the training program can estimate these slowly varying attributes.

3.3.5 Estimating Personality, Cognitive Styles, and Learning Preferences

These attributes usually vary slowly and can be estimated by test batteries administered before or during the training program if there are known correlations between the test measurements and the personality, cognitive style, or learning preference attributes.

The AIS can re-assess the set of learning preferences assumed for the student if the student's actual learning rate falls short of the expected rate, based on normative and individual determinants of the learning rate. When the AIS hypothesizes that the student's learning preference may be different than was previously assumed, it can design and perform "experiments" to test whether the student has a different learning preference by administering the learning method that is compatible with the new, hypothesized learning preference and checking whether or not the new learning method yields improved results.

3.4 Instructional Planner

Most ITSs use simple algorithms to select instructional decisions based upon relatively few attributes of the student. An AIS, however, will consider many more attributes and will pursue many more instructional goals to optimize the student's learning and retention. This level of sophistication will require the AIS Instructional Planner to make instructional decisions by processing explicit knowledge representations of instructional goals and plans. Automated planning is another classical discipline within artificial intelligence.

3.4.1 Instructional Goals

The AIS has two primary instructional goals:

- select and provide learning experiences to the student that result in effective learning, and
- assess the student's skills, knowledge, and other student model attributes to provide a valid basis on which to select learning experiences.

To pursue these high-level goals, the Instructional Planner will create and pursue more specific, lower-level goals, such as presenting a practice experience for skill c increasing/decreasing the student's stress-level to an optimum level, or estimating the student's proficiency in skill S_2. High-level goals may be pursued over longer periods, spanning the length of the training program or perhaps the student's entire career. Other goals may be short-term and possibly identified and pursued opportunistically during the execution of a training task.

3.4.2 Instructional Decisions

To achieve these primary instructional goals, the AIS makes Instructional Decisions to select and configure tasks or exercises for the student to perform, to provide learning experiences to the student or to provide diagnostic information about the student for the AIS. There are many generic strategies that can be employed, such as:

1. If the student is weak in a skill, present the student with opportunities to practice tasks that employ the weak skill or are otherwise effective ways of improving those skills,
2. If the student is weak in a skill, create a goal of determining whether a possible cause might be a subskill with low proficiency or automaticity,
3. Provide consistent practice opportunities to help the student learn a skill. For example, the AIS could provide the student with repeated opportunities to practice hovering under similar conditions,
4. To minimize over-specific learning and widen the student's "performance envelope", vary parameters of the task. For example, after the student achieves moderate proficiency performing a hover task, the AIS could vary the position and heading of the hover task to prevent the student from relying upon incidental visual cues that are present only in the original hover task,
5. Consider learning decay effects to determine the frequency with which practice opportunities for a task should be repeated,
6. Provide the student with an appropriate level of challenge to the student, based upon the student's proficiency in the relevant skills,
7. Maintain the student's confidence by presenting tasks that are within his or her ability to succeed,
8. Select tasks for the student in order to obtain diagnostic information that helps the AIS estimate student model attributes. For example, to assess the proficiency of a skill, select a task that results in the assessment of the skill if such a lesson exists,
9. To detemine whether a weak subskill is the cause for the poor performance of a task, compare the task performance with the performance of a second task that eliminates the need for the subskill. For example, a student may be having trouble with hovering because he or she cannot gauge position accurately using normal visual cues. A

diagnostic task might have the student hover with the aid of a graphical display that simplifies the gauging of xy position. If performance of this diagnostic task is significantly better than performance of the original task, we can infer that gauging xy position was a cause of the poor hovering proficiency,

10. Provide demonstrations and hints to help the student successfully execute a task,
11. Provide quantitative feedback for tasks with quantitative performance measures, in order to accelerate learning (Knowledge of Results principle),
12. If the student is busily engaged in the task, provide hints using short phrases and other methods of minimizing the cognitive load incurred by the student when listening to feedback.

Like any goal-directed system, the AIS Instructional Planner must pursue these goals within resource constraints. For instructional systems, the primary resource limitation is the student's time and attention, so the Planner must prioritize the goals it achieves and strive to achieve multiple goals simultaneously, when possible.

Different decisions are made at different times. For example, some decision about the content and format of feedback provided during lessons might be made continuously during the lesson. Selection of the next lesson occurs only after the previous lesson completes.

3.4.3 Adapting Instructional Decisions to Individual Differences

The AIS can use student attributes estimated within the student model to help:

- Select and prioritize instructional goals that the AIS should pursue with respect to the student, and
- Select plans (make instructional decisions) that pursue these instructional goals effectively for the specific student.

Examples of adapting instructional decisions to individual differences include:

- Attempting to increase the student's automaticity for a skill A (i.e., by providing practice opportunities) if it appears to be the cause of poor proficiency with another skill for which skill A is a subskill.
- Explaining task steps or cause-and-effect relationships in more detail if a lack of understanding of these steps or relationships appears to be the cause of poor skill proficiency.
- Setting the difficulty or target performance level, based upon the student's proficiency and the student's personality traits (e.g,. the effect of a successful or unsuccessful task execution on the student's confidence and motivation).
- Modifying the task to adapt to student strengths and weaknesses in subskills of the task. For example, weak perceptual skills may be preventing the student from learning the procedural and motor-control aspects of the task. The AIS might present cognitive aids that simplify the perceptual aspects of the task, to help the student focus on learning these other aspects of the task.

4. Augmenting the IFT with AIS Capabilities

The Intelligent Flight Training (IFT) is a helicopter flight simulator that is augmented with automated instructional capabilities that 1) modify the simulator's flight dynamics based on the student's proficiency and 2) provide verbal instructional feedback to the student during simulated flight. IFT is installed at the Army Research Institute at Ft. Rucker, AL.

4.1 Instructional Strategies Used by Human Instructor Pilots

Learning is facilitated when the student is able to achieve "meaningful repetitions" of the maneuvers. Instructors achieve this by:

- Clearly explaining and demonstrating maneuvers beforehand,
- Applying building-block instructional strategies, so students learn simple maneuvers which are pre-requisites for complex maneuvers. Complex maneuvers can combine simple maneuvers (e.g., climb and turn) or are carried out under more difficult conditions (e.g., with wind, landing/takeoff on sloped hill),
- Providing hints and feedback before and during the maneuver, to increase the likelihood of a successful execution.

According to instructor pilots at Ft. Rucker, "80% of helicopter training is psychology." Instructors select maneuvers and provide verbal feedback to increase the student's confidence and reduce his anxiety. For example, instructors can provide positive feedback when the student performs a maneuver (or portion of a maneuver) correctly. This, of course, requires the instructor (or automated instructional system) to have a fairly accurate model of the limits of the student's skills, so that the instructor can notice when the student has progressed beyond these limits.

At the same time, instructor pilots must ensure that students are motivated and challenged, to maximize their rate of learning. This entails selecting maneuvers and performance targets that are at the edge of the student's skill proficiency. For example, the instructor would provide instructional feedback that presumes successively higher expectations for performance levels as the student advances through training.

Instructor pilots adapt their instruction to individual differences among students by varying:

- the rate at which they progress through the training maneuvers,
- the academic exercises they assign,
- the amount of direction they provide to guide each student's studies,
- inter-personal style, and
- rate of speech.

The remainder of section 4 describes methods of applying AIS capabilities to improve automated helicopter pilot training.

4.2 Selecting and Configuring Helicopter Piloting Tasks

An important and challenging type of instructional decision is selecting and configuring tasks for the student to carry out. At any point in time, the instructional system will have many instructional goals, where each goal support the selection of a task in order to:

- Provide an opportunity for the student to practice a task for which his or her proficiency or automaticity is low,
- Measure the student's performance carrying out a task to estimate the student's proficiency or automaticity for the task or subtasks.

Task configuration is thè modification of attributes of the task to achieve additional instructional goals. For example, the initial position; target speed, direction, or heading; or wind velocity can be modified to minimize over-specific learning. Auxiliary, concurrent tasks can be added to estimate the student's skill automaticity, by measuring the student's performance when some of his or her attention is occupied by the concurrent task.

4.3 Setting Target Performance Levels

Students require targets levels of performance to strive for. Ideally, these targets are achievable, but require the student to try hard. In the domain of helicopter pilot training, it is possible to set target performance levels as tolerances on flight variables such as altitude, speed, heading, and drift that are matched to the student's proficiency level.

4.4 Selecting Hints and Feedback

The AIS can present hints before the student attempts a maneuver to remind the student of important concepts, to increase the likelihood that the student achieves a meaningful repetition. The AIS can present visual and/or verbal feedback to help the student learn the relationships between control movements and the aircraft's response. Visual aids display the state of the aircraft in ways that make it easier for the student to assess certain aircraft states. This learning scaffold can support both practice and diagnostic goals. For example, if it is desirable for the student to focus on practicing motor control movements, visual aids can simplify the perceptual task so that the student can apply his or her full attention to the motor control task. Visual aids also help diagnose student difficulties. For example, if the student performs significantly better with the aid than without it, the AIS can infer that the perceptual task simplified by the visual aid is either not yet learned or not yet overlearned.

5. Phase I Software Prototype

We developed a limited, proof-of-concept software prototype to test, refine, and demonstrate some of our ideas for the Instructional Planner portion of the AIS.

The prototype:

1. Reads a user-selected file containing the desired student model,
2. Selects the next lesson (or task) by:
   - Generating instructional and diagnostic goals,
   - Proposing candidate lessons that achieve these instructional and diagnostic goals,
   - Selecting the best lesson,
3. Adapts the lesson to other student attributes by:
   - setting target performance levels to levels achievable by the student, given his or her proficiencies,
   - adds auxiliary tasks to the primary task, to test automaticity,
   - Sometimes displays visual aids to simplify subtasks of a task for diagnostic or instructional purposes,
4. Presents the lesson briefing to the student that describes the student's task and performance objectives.

This prototype is composed of two programs that communicate via DirectX inter-process communication, as shown in Figure 4.

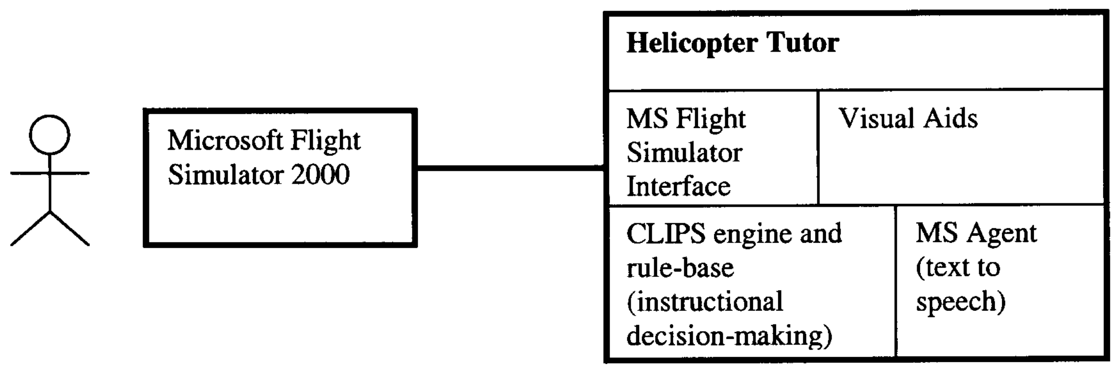

Figure 4 - Architecture of the software prototype

- **Microsoft Flight Simulator 2000.** This Microsoft software product lets the user pilot a simulated aircraft. The user controls the aircraft using a joystick and generates a cockpit view with an instrument panel. It simulates more than a dozen aircraft, including a Bell 206B helicopter. A dial and a twistable handle on the Microsoft joystick emulate the pedal, cyclic, and collective controls.
- **AIS Tutoring System.** This prototype C++ program developed by SHAI for this project receives flight variable values from Microsoft Flight Simulator using its DirectX interface. The tutor embeds a Microsoft Agent object to generate speech from text, to talk to the student and to explain its reasoning as it selects and configures a task for the student to perform. The CLIPS rule engine, developed by NASA, runs the tutoring system's rule base that implements the instructional decision-making.

The student runs each lesson by controlling the computer's joystick to execute the task. The Helicopter Tutor receives notification of the aircraft's flight variables, updates the visual aid if it is displayed, and forwards these flight variable values (along with some derived values) to the Clips rule base. The rule base provides verbal feedback to the student delivered via MS Agent speech-to-text, considering the target performance levels selected for the student and task.

Figure 5 shows the user interface presented by the prototype.

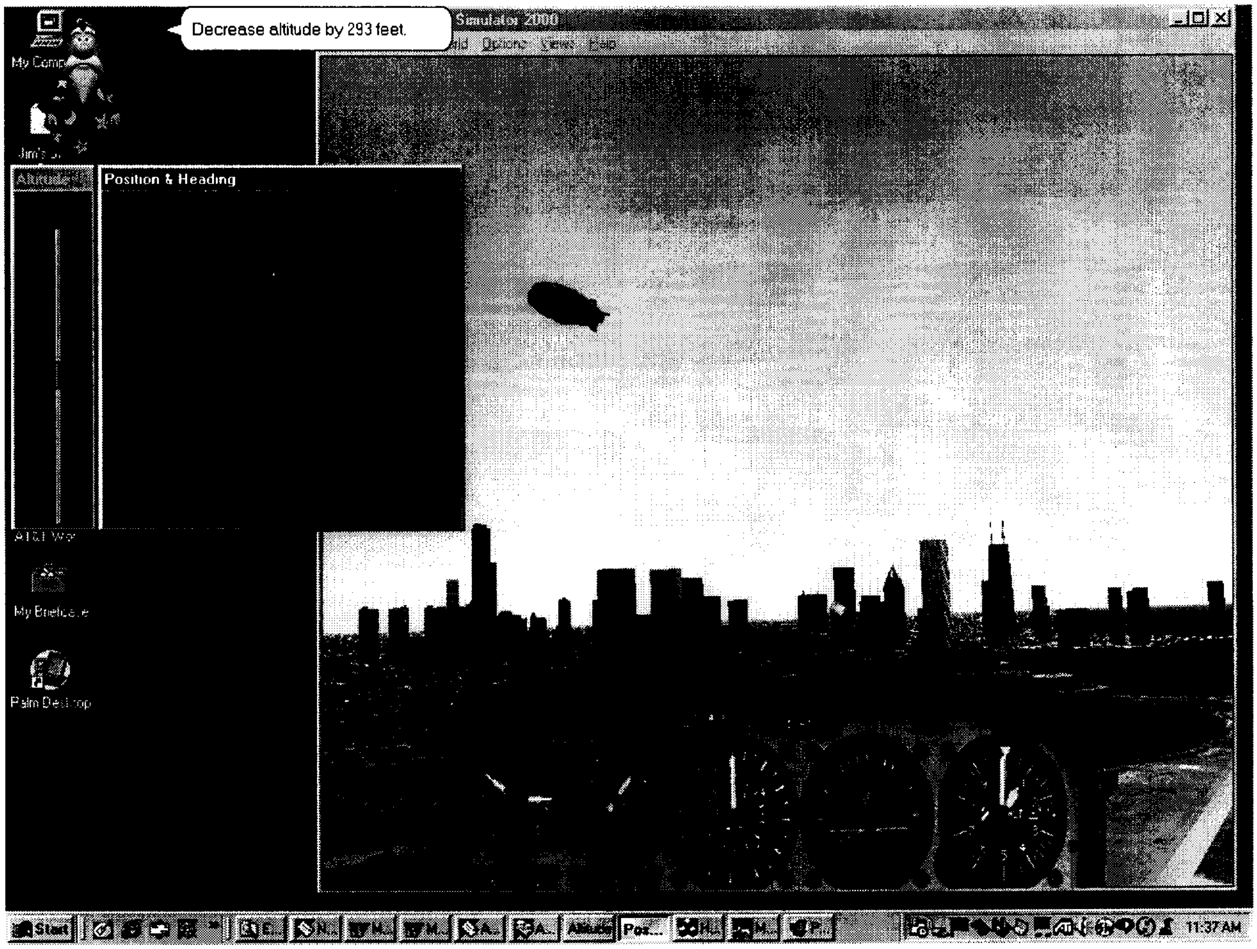

Figure 5 - Screen capture of AIS software prototype with MS Flight Simulator cockpit view window (at right) and hover task visual aid (at left). Feedback is delivered via text-to-speech.

The cockpit view window displayed by MS Flight Simulator is on the right portion of the screen, and two windows that comprise the hover visual aid are on the left portion of the screen. The left hover aid window shows the current altitude as a short red vertical bar, the standard tolerance as a blue vertical bar, and the target tolerance selected for this student and task as an overlaid green vertical bar. The target tolerance may be wider than the standard tolerance if the student is not yet proficient in maintaining altitude. The right hover aid display window shows the corners of a blue rectangle corresponding to the task's standards for drift, and it displays the rectangle corners corresponding to the target tolerances for drift in green. The helicopter is shown as a yellow icon comprised of a circle which represents the cockpit (indicating the aircraft's xy position) and a line that represents the fuselage (indicating the heading).

The Helicopter Tutor relies on the following knowledge to base its instructional decisions:

- The student's proficiency and automaticity level for each skill,
- Expectation levels for proficiency and automaticity, based upon each student's number of weeks of training,
- Hierarchical relationships between skills and subskills,
- Lessons associated with skills that x increase proficiency or automaticity,
- Associations between skills and lessons that help the student increase their proficiency in those skills,
- Methods of modifying lessons to achieve additional instructional goals, such as setting appropriate target performance levels and estimating automaticity levels of subskills.

For example, if the student's proficiency in a skill, S_1, is lower than expected for students with the same amount of training, the Tutor creates a goal of increasing the student's proficiency in that skill as well as understanding the underlying cause(s) of the low proficiency. A possible explanation for the low proficiency might be that the student's proficiency or automaticity in a subskill of S_1 is low. Thus, the tutor searches for any estimates of proficiency and automaticity of subskills that are either low or unknown. If the estimate is low, the Tutor creates a goal of increasing the automaticity and/or proficiency of the subskill. If the estimate is unknown, the Tutor creates a goal of estimating the subskill's automaticity or proficiency. The Tutor then evaluates possible lessons that achieve the various instructional goals it has created and selects the lesson that achieves the most important goal. The Tutor then modifies the task to achieve additional goals or to adapt it to the individual. For example, the Tutor will set target performance levels for the student, based on the student's proficiency levels. Students with low proficiency levels for a skill would be given appropriately low performance targets, such as wide drift tolerances when hovering. Cognitive aids such as the hover visual aid would be enabled or disabled, depending upon the task, the student's proficiency, and the instructional goals for the student. The Tutor adds additional subtasks to be carried out concurrently with the main task to estimate skill automaticity or provide practice to increase automaticity.

6. Proposed Phase II Work

We propose to use the AIS model developed during this project as a starting point for further research in AIS capabilities in other domains, such as command and control. Specifically, we propose to continue this research using Phase II SBIR funding to add adaptive instructional capabilities to two ITS systems developed or under development by SHAI. First, SHAI will enhance the Tactical Action Officer (TAO) ITS which is described in section 8.1.1. The TAO ITS is an existing intelligent tutoring system developed by SHAI for the U.S. Navy that teaches tactical decision-making and command and control principles. Second, we will also extend a prototype C4I ITS intelligent tutoring system currently under development by SHAI for STRICOM that interfaces with the Force XXI Battle Control Brigade and Below (FBCB2) command and control system, as described in section 8.1.2. We will extend the functionality of the C4I ITS prototype and incorporate adaptive instructional capabilities.

We expect that the high-level AIS model described here will apply to these command and control domains. However, this model will require significant elaboration to guide the software design of any domain-specific ITSs. Specifically, we will need to:

- Identify the exact student attributes to be modeled in these ITSs, based upon their usefulness and ability to be estimated in these specific domains,
- Identify and develop generic and domain-specific Student Assessment algorithms, and
- Identify the set of generic and domain-specific instructional strategies to be implemented within the Instructional Planner.

We believe that these domains are excellent test beds for research in adaptive instructional capabilities. We plan to explore the feasibility and usefulness of estimating and considering personality, decision-making style, and general computer skills proficiency when making instructional decisions.

7. Phase I Option Results

7.1 MAMID

7.1.1 A Summary of Methodology for Analysis and Modeling of Individual Differences (MAMID)

The following summary is derived from work by Hudlicka and Billingsley (1999), entitled *Representing Behavior Moderators in Military Human Performance Models*. The tables included in this summary are directly from their work as well.

Introduction

The motivation for this work is the increasing amount of resources being spent by the military on computer based simulation and training programs, and that these programs are limited by their inability to represent and express "normal" behavior variations. They briefly discuss the idea of individual differences (cognitive, personality, & affective) and provide a few examples of possible relationships between individual difference factors and performance in Table 1.1-1. The implicit assumption is that if agents in training or simulation programs could express such individual differences, these programs would be more realistic and effective.

Table 1.1-1: Possible Effects of Cognitive, Affective, and Personality Factors on Behavior

Maintain excessive reserves due to low-anxiety tolerance, making mobility more difficult
Fail to recognize a situation as distinct due to overly assimilating cognitive style, limiting further actions
Falsely interpret tank as enemy due to heightened anxiety, risking possible fratricide
Fail to react to a warning signal due to high risk-tolerance, risking lives of personnel
Select/ avoid a particular maneuver due to individual experience & training, rather than task requirements
Overreact to ambiguous intelligence reports due to anxiety and low risk tolerance
Prefer planning based on historical cases, due to a bias for case-based reasoning, limiting maneuver options
Prefer goal-directed reasoning, increasing likelihood for confirmation bias and false situation assessment

Hudlicka and Billingsley developed MAMID to address the problem of representing individual differences in models of human behavior. Their methodology is based on the four steps, as seen in Table 1.2-1. The enhanced models of human behavior derived from this work could then be used in computer based simulation and training programs. They go on to develop their ideas in the context of an Army combat scenario.

In this exercise, they point out how traits such as *low anxiety tolerance* and *high preference for case-based reasoning* might affect a tank platoon commander. They also cite relevant research that connects these individual differences to performance.

Table 1.2-1: Methodology for Analysis and Modeling of Individual Differences

METHODOLOGY STAGE	EXAMPLE
1. IDENTITY DISTINCT COGNITIVE PROCESSES & STRUCTURES MEDIATING SKILLED PERFORMANCE	**Processes:** Attention, Perception, Inferencing (e.g., Situation Assessment, Decision Making) **Structures:** Short & long term memory
2. DESIGN A CORRESPONDING PARAMETERIZED COGNITIVE ARCHITECTURE	**Architecture Modules:** Perception, Situation Assessor, Decision Maker, Attention, Memory , etc. **Controlling Parameters:** Thresholds for matching internal models & incoming data; Speed of processing, etc.
3. IDENTIFY COGNITIVE, AFFECTIVE, AND PERSONALITY FACTORS AFFECTING INDIVIDUAL MODEL PROCESSES	**Cognitive:** skill level, cognitive style, bias susceptibility **Affective:** current affective state, temperament (e.g., anxiety & risk tolerance) **Personality:** introversion/extraversion, leadership style, negative/positive emotionality
4. ENCODE IDENTIFIED FACTORS IN TERMS OF MODEL PARAMETERS & KNOWLEDGE BASES Different parameter values induce different model behaviors, allowing for implementation of a range of behavioral variations due to individual profiles (e.g., cognitive, affective, personality, and morale factors)	**Attention:** anxiety level affects scan speed, selectivity, ease of engagement/disengagement **Perception:** skill & anxiety levels affect perceptual processing and situation assessment **Memory:** mood state affects retrieval; skill level & training affect content & organization **Inferencing:** affective state affects speed & decay of activated units; skill level & cognitive style affect preferences for particular inferencing types

Description of MAMID

The cognitive architecture of MAMID is developed and discussed in detail in their paper, but can be summarized as a "modular cognitive architecture" which makes use of a variety of knowledge representation formats, such as Bayesian belief nets and rules. The main components are the *attention, situation assessor, decision selector, and procedure executor and monitor* modules.

Module behaviors are controlled by the contents of the knowledge base and the *processing control parameters*. To bias the module towards a particular behavior, the related processing control parameter would be altered. A list of possible parameters is given in table 4-1.

Table 4-1: Cognitive Architecture Parameters and Features Capable of Modeling Effects of Individual Differences

Attention
- Scan speed
- Ease of engagement / disengagement
- Scan intensity (degree of focus)
- Selectivity

Memory
- *Content*
  - Degree of conceptual complexity and differentiation
  - Type and size of memory units
- *Retrieval*
  - Speed and accuracy of retrieval
  - Divergent vs. convergent search
- *Organization*
  - Type of internal structure (e.g., hierarchy, causal model, etc.)
  - Level of interconnectivity among knowledge units

Perception
- Specificity of correspondence required between input data and stored category
- Speed of detection and matching processes

Inferencing
- *Generic*

Speed of inferencing Decay of activated units *Specific* Meta-cognitive inferencing What-if simulation Causal analysis Data-driven vs. goal-driven processing Recall vs. derivation of required data Movement between multiple levels of abstraction

They then provided a summary of cognitive, personality, and affective parameters, which they found to be supported by the literature as likely to influence behavior. These are summarized in Table 5-1.

Table 5-1: Examples of Individual Differences Factors Influencing Performance

Personality / Affective
Big 5
Extraversion
Emotional stability
Agreeableness
Openness
Conscientiousness
"Giant 3"
Approach behaviors
Inhibition behaviors
Aggressiveness

Affective State
Anxiety
Fear
Sadness (low mood)
Boredom
Alertness
Happiness, Joy
Anger
Surprise
Disgust
Guilt
Shame

Cognitive
Generic abilities
Working Memory (WM)
Capacity
Speed
Accuracy
Long term memory
Valence of recalled material
Speed
Accuracy
Attention
Speed
Accuracy
Capacity
Vigilance
Specific abilities and skills
Wargaming
What-if reasoning
Specific topics / skills
Stylistic factors
Cognitive bias susceptibility
Visual vs. linguistically-oriented
Assimilating vs. accommodating
Analytic vs. intuitive
Goal directed vs. data-directed
Case-based vs. 1st principles reasoning
Individual history
Preferred / Avoided situations
Preferred / Avoided maneuvers
Preferred / Avoided target types
Previous successful / failed operations
Training / Education
Doctrinal emphasis on sp. maneuvers
Doctrinal emphasis on particular unit
Doctrinal preferences for timing
Specific area of decision-making competencies and vulnerabilities

At this point, they have provided both a set of model parameters and a list of individual difference parameters. The problem now is to connect these individual difference parameters to the cognitive architecture parameters, and to define the domain specific ways in which the cognitive architecture parameters affect behavior in the simulator. A few examples of relationships (such as anxiety tolerance -> bias towards threats) are presented, but for the most part detailed descriptions of these relationships are not given.

Conclusion

A version of the MAMID was being developed within a computer simulation to demonstrate the effectiveness of this model in the tank scenario discussed above. The current status of this work is currently unknown. They summarized the goals of their paper as:

- To represent a particular individual's behavior by capturing his/her cognitive and decision-making style, and
- To make simulated training exercises more realistic by providing for a range of behavior for both friendly and opposing forces.

Finally, they pointed out some of the limitations of this work such as problems assessing individual differences and that it is not know to what degree individual differences change over time or in different domains.

References

Hudlicka, E., & Billingsley, J. (1999). Representing behavior moderators in military human performance models. In *Proceedings of the 8th Conference on Computer Generated Forces and Behavioral Representations.*

7.1.2 Role of MAMID in an Adaptive Instructional System

Motivation

Currently, the student model used in the tutoring system does not take affective state (physical, emotional, mental), learning style, cognitive style, or personality factors of the student into account during instruction. Elements of MAMID could be very useful in augmenting a student model to take advantage of this information. Three different types of actions could be performed based on knowledge of individual differences: adaptation of instruction, providing guidance, and informing the student.

Use

The first type of action is adaptation of instruction. This would involve altering the teaching style or content of the instructional system to take into account factors specific to the individual. Examples of this for learning styles can be seen in the learning styles section of this report. An example of altering content might be tending to retrieve less challenging scenarios for more anxious students; retrieving scnearios where boldness is required for more timid students (thus practicing their weaknesses); or use of specific remedial exercises to address the particular individual factors, if they are a detriment.

A second type of action would involve guiding a student to the correct solution. The idea here is that if a student is likely to make a mistake, it will enhance learning to proactively guide her in the right direction rather than letting her make a mistake and providing a subsequent remediation. A relevant example given by Hudlicka and Billingsley is that of having high anxiety in an ambiguous situation. In this context, the student is likely to perceive ambiguous information as a threat and may react prematurely. Rather than letting this happen for an anxious student, an AIS might proactively suggest that the student wait for more information before acting, thus preventing the mistake.

Another use of information on individual differences would simply be to inform the student of their strengths and weaknesses. For example, if a student knows that they usually behave too timidly, they could mentally compensate for this deficiency when deciding what course of action to pursue.

Limitations

A primary limitation of using MAMID is the assessment of individual difference factors in each student. Many of the factors described in the model would be difficult, if not impossible, to accurately assess for every student that uses the AIS. Another related problem is the extent to which the individual differences will transfer to the training domain, or across different training domains.

The model developed by Hudlicka and Billingsley was designed to provide realistic behaviors in a software agent, not to predict behaviors of humans using simulation software. In the above example given about anxiety and perception, it must be realized that a student would have a different belief system than a software agent. While the agent "believes" that it might die (hence the anxiety), the student might be more anxious about using the adaptive instruction system than the simulated threat. So the correct action for the AIS in this situation might be to work slowly with the student to make her feel more comfortable with the system. To alter the model to correct for the different belief systems will require significant work on assessing the relationships between individual difference factors and model parameters for students using a simulation, similar to the work done on these relationships for software agents.

Finally, the model as presented would require a large amount of background knowledge in order to determine the likely interactions between individual differences, model parameters, and behavior. This would likely involve developing an expert system for each domain.

Conclusion

Despite the limitations of the model discussed above, the MAMID methodology still looks to be quite useful in constructing an AIS. It provides an outline of the tasks which need to be completed to effectively use individual differences in the context of an adaptive instructional system.

7.2 Learning Styles

Introduction to Learning Styles

A motivation of adaptive instruction is the idea that all learners do not learn the same, and that an increase in students' performance could be realized by teaching to their individual differences (Federico, 2000). One way of adapting instruction to an individual is through altering the instructional style to match the student's learning style. Federico purports that by understanding a student's style, a teacher could: *"improve the planning, producing, and implementing of educational experiences, so they are more appropriately compatible with students' desires, in order to enhance their learning, retention, and retrieval"* (2000, p. 367).

The simplest definition of a learning style (also called a cognitive style) is the learning strengths and preferences of a student. However, many other definitions exist, such as the one given by the National Association of Secondary School Principles. They define a learning style as *"the composite of characteristic cognitive, affective, and physiological factors that serve as relatively stable indicators of how a learner perceives, interacts with, and responds to the learning environment"* (Keefe & Monk, 1986, p 1). Despite the plethora of definitions for learning styles, the basic idea can be seen through a few examples provided by Felder (1996). He discusses that some students might focus on facts while others prefer theories or that some students learn better visually and others verbally. Similar to the large number of definitions for learning styles, there is no one accepted model of, or assessment for, learning style (Hickcox, 1995).

Curry (1987) tried to organize the learning style field, and defines four areas of learning styles in an "onion" model: instructional preference, information processing factors, social interaction preference, and personality factors. It is called an onion because the outside layers (instructional preference) change over time, whereas the inner layers (personality factors) are stable over time. Of these four elements, instructional preference, information processing characteristics, and personality factors are likely to apply to the adaptive instructional system (AIS) as described in this report. Unfortunately, this does not simplify the problem of deciding which learning style model (and accompanying inventory) would be best to use in an AIS. Leslie, Perry, and Landrum (2000) provide a concise summary of learning style models that fall under these three headings. Under instructional preference they list theories by Canfield and Dunn and Dunn. Models by Gardner, Gregorc, and Kolb are classified as information processing models. Finally, Meyers-Briggs and Witkin models are described as personality characteristics models. The Felder-Silverman, Rider, O'Brien, and Keefe and Monk models are four additional examples not discussed by Leslie, Perry, and Landrum.

Models

<u>Instructional Preference Models</u>

Dunn & Dunn's (Dunn, 1984) model is based on five environmental and instructional preferences. The first of these preferences is environmental, which covers sound, light, temperature, and class design. Emotional preferences contain motivation, persistence, and structure. The sociological preference addresses learning alone or in groups, as well as learning relationships. A physiological preference element contains such things as perception, time, and mobility. The final element contains psychological preferences, of which hemesphericity or analytic mode are good examples (O'Connor, 2000). The student could be ranked in these dimensions by using the Productivity Environmental Preference Survey, developed by Price, Dunn, and Dunn. (Murray-Harvey, 1994).

The Canfield model (1980) was based on four learner scales: *conditions of learning*, *content*, *mode*, and *expectation*. Conditions of learning might include affiliations, structure, and orientation towards goals and competition. Numbers and language would be classified as content. Mode captures a student preference for listening, reading, iconic, or direct experience, while expectation is related to the expected grade. The two inventories for this model are the Canfield Learning Style Inventory (CLSI) and the Canfield Instructional Style Inventory (CISI). While the CLSI was created to assess learners, the goal of the CISI is to help students and faculty communicate about learning activities (Swanson, 1995).

<u>Information Processing Models</u>

Gardner provides a theory of multiple intelligences. The seven different types of intelligence are linguistic, logical/mathematical, spatial, musical, kinesthetic, and inter/intrapersonal. The focus of this theory of multiple intelligences is on the content of the material to be learned and it's relationship to various disciplines. A limitation given of this model, with relationship to learning styles, is that the process of individualized learning is not addressed (Silver, Strong, & Perini, 1997).

Kolb's model classifies students on two dimensions: concrete experience (CE) or abstract conceptualization (AC) and active experimentation (AE) or reflective observation (RO). Using this model, students are classified into one of four types based on how they perceive information (CE/AC) and how they learn information (AE/RO). This theory takes into account that people could use any of the four styles some of the time by claiming that the classification is a preferred method, not an exclusive one. Kolb also developed a Learning Style Inventory (LSI) to categorize students according to this model (Willcoxson & Prosser, 1996). A more recent version of the inventory is titled the LSI-IIa (Smith & Kolb, 1996).

Gregorc's (1982) model is similar to Kolb's, except that the two dimensions rate perception from abstract to concrete and ordering from sequential to random. The final classification of the learner is into one of four states, again similar to Kolb, using the Gregorc Style Delineator.

Personality Factors Model

Witkin's model is a bipolar construct. The two ends of the spectrum are field dependence and field independence, which relate to how much a learner is influenced by the surrounding field. People would not fall into one of two categories, but would rather be placed somewhere in the continuum between the two poles. Witkin developed the Group Embedded Figures Test to classify individuals on this construct. Basically, the test required people to find a simple object in the context of complex objects. Those classified as field independent would be able to perceive the figures, despite the complexity of the surrounding field (Swanson, 1995). This type of test has been used to select people for positions that require selection of objects from a complex field such as pilots and bus drivers.

The Myers-Briggs Type Indicator (MBTI) defines sixteen different personality types through the use of four factors. The factors used by this model are *extraversion* (focus on people) / *introversion* (ideas), *sensors* (detail oriented) / *intuitors* (imagination oriented), *thinkers/feelers*, and *judgers/perceivers* (Felder, 1996). Felder also describes some examples of students' classifications and their associated learning blind spots. The MBTI usually consists of around 90 questions, takes 15-25 minutes to complete (Consulting Psychologists Press, 2000), is paid for on a per-test basis, and requires a qualified person to proctor and interpret the test. An online version of the test is currently priced at $99 per test (Knowyourtype.com, 2000). The Keirsey Temperament Sorter is a test that is freely available online and is correlated with, but not a substitute for, the MBTI (Kiersey, 2000). Despite the large number of MBTI users, there are some reservations about its reliability and the fact that determining learning style is not one of its recommended uses. Additionally, the MBTI is not necessarily valid across different domains (Noe, 1999). For example, a study by Cooper and Miller (1991) found that students were primarily *sensors* while professors taught in an *intuitive* style, but did not find that in-class teacher/student congruity led to higher grades.

Other Models

The Felder-Silverman model seems to contain aspects of both instructional preference and information processing. The five dimensions of this model are: sensing/intuitive, visual/verbal, inductive/deductive, active/reflective, and sequential/global (Felder, 1996). The Index of Learning Styles inventory would classify students in all of the categories except inductive/deductive. A free online version of the test is available at:http://www2.ncsu.edu/unity/lockers/users/f/felder/public/ILSdir/ilsweb.html. The test contains 44 questions, which could be roughly estimated to take 10-15 minutes.

Another model that classifies learners on two continious dimensions is the model by Riding (1997). The first dimension is *wholist/analytic*, which describes if a person processes information in wholes or parts. *Verbaliser/Imager* is the second dimension, which describes how a learner represents information verbally or pictorially. Rider discusses various tests by others for measuring these traits, as well as the Cognitive Styles Analysis (CSA), which he had developed previously. The CSA is a computer-based test which scores students on these two dimensions.

O'Brien (1989) discusses a model of *learning channels*, which describes three modalities of learning: auditory, visual, and haptic. He claims that learning channels affect both how students learn and how they demonstrate what they have learned. He provides a Learning Channel Preference Checklist for use in classifying an individual according to their preferred learning channel.

The last model discussed here is that presented by the NAASP (Keefe & Monk, 1986). They describe a comprehensive Learning Style Profile, which determines students cognitive, affective, and physiological styles. Examples of cognitive style might be analytical or spatial skills, persistence or grouping preferences would be part of affective styles, and perceptual response and study time preference are examples of physiological styles.

Which model to use?

In order for a learning style model to be useful in an AIS it must meet the following requirements:

1. The inventory to assess the learning style of the student should be quick, easy, and inexpensive to perform as well as statistically valid.
2. The model should be generally supported in the research literature.
3. The model must contain dimensions that can be used to successfully adapt the course to improve the learner's performance.

The models discussed above by O'Brien, Canfield, Dunn and Dunn, Felder-Silverman, and Keefe & Monk all contain a perceptual aspect, sometimes called a learning channel. Kirby, Moore, and Schoefield (1988), not discussed above, is yet another example of research on this topic. Most learning channel theories describe learners as visual, auditory, or kinesthetic and propose that students will learn best when presented with information in their preferred modality. The learning channel model meets the easy and inexpensive requirement. A learning style test is available on the Internet for a nominal charge. This test contains 30 items and would probably take between five and ten minutes to complete. While statistics for this test are currently unavailable, it does have the benefit of being actively used by several large corporations and government branches (Center for New Discoveries in Learning, 2000). Additionally, since the learning channel model is part of several of the competing learning style models, it seems to meet the second requirement of general support. How this model meets the final requirement will be discussed below.

The Kolb model is another model that could be used within an AIS. The LSI test is described by (Newstead, 1992) as quick and easy, but he does question the tests validity. The LSI2 test for Kolb's model (described above) was created to address validity problems such as these. This model has much in common with the Gregorc and Felder-Silverman models, lending to it being acceptable as generally supported. The possible uses for this model in an adaptive instruction system will be discussed below as well.

Finally, the personality model proposed by Myers-Briggs could also be used. Since the questionnaire takes around 25 minutes, and seems to be expensive, it pushes the limits for quick, easy, and inexpensive. Despite reliability questions, it is currently widely used, which meets the second requirement. Regarding the third requirement, it seems intuitive that the results of this personality measure can be used to effectively instruct the student, though at this time we have not found research describing what are the right connections to make between personality type and effective teaching strategies.

However, because of the diversity and number of learning models, there will inevitably be other models that would be relevant or interesting. Therefore, an AIS should allow the content author to develop additional learning style models. Furthermore, since learning styles can be domain dependent, the author needs to be able to develop domain specific interactions between learning style and content. A good example of the need for this adaptability is the concept of *boldness*. This concept is not part of any of the examined models, but might be especially important for an AIS system providing military training. This adaptability also creates the added benefit of allowing an AIS to be used as a tool for further research in learning styles.

Practical Use of Learning Style Models

McLoughlin (1999) discusses the idea of using a learning style model for adaptive courses. These steps neatly match a single phase of an AIS:

1. Identify learner specifics
2. Select and organize content
3. Define the pedagogical profile of the adaptation
4. Develop the instructional unit-pedagogical profile, media, and materials

The first step is to assess the learning style of the user, as described above using a learning channel test. In the second step, the knowledge of the users learning style would be used to select and organize content.

<u>Learning Channel Model</u>

The Armor Captains Career Course (2000) provides several suggestions for taking advantage of learning channel knowledge in presentation of course material (Step 2). Some of the ideas that could be adopted by an AIS are:

- For students that are visually oriented
    - Provide notes in an outline form
    - Provide a way for them to develop outlines when attempting to answer complex questions
    - Make use of extensive visual association and imagery
    - Visually cluster ideas for the student
    - Provide a method for the student to perform written repetition. An example of this might be to provide a *notepad* for them to type in information they want to keep at the end of a session.
- For students that are auditory oriented
    - Encourage the student to rehearse information orally or to use subvocalization
    - Eliminate extemporaneous noise (mouse clicking noises, etc.)
    - Provide audio in place of text. For example, add audio sound bytes to diagrams rather than text
- For students that are kinesthetic oriented
    - Keep auditory information to a minimum
    - Encourage active participation through hands on activities. An example of this would be an interactive training "game" to teach complex ideas

- Present information organized into steps necessary to complete the task
- Provide summaries for the student for complex text

Kolb's Model

Several ideas for using Kolb's model discussed by Felder (1996) could also be implemented in an AIS. The basic tactic is to alter the teaching style to suit the student. Here are the four styles and their associated teaching style:

1. Concrete/Reflective – This type of learner needs to be *motivated.*
2. Abstract/Reflective – This type of learner needs an *expert.*
3. Abstract/Active – This type of learner wants a *coach.*
4. Concrete/Active – This type of learner wants to think for herself (*advisor*).

So, to take advantage of Kolb's model the AIS must be able to act in all of these four modes. A motivator would explain the relevance of the material and relate it to the student's experience. An expert would focus on explaining the information in an organized fashion and allow time for reflection. The coach model would promote learning by trial and error by providing guided practice and feedback. The final mode, which will be referred to as the advisor, would provide new and complex real world problems and generally allow students to have the chance to discover things for themselves.

Myers-Briggs Model

The MBTI would provide a ranking of an individual on one of the four preference scales discussed above. Intuitively, this model seems like it could be useful in adaptive instruction. For example, imagine a student rated highly on the *judger* scale and another rated high on the *perceiver* scale. In a tactical situation, a judger might seek to make a judgment with incomplete data. A training system that knew this in advance could give a hint to the user that they should wait a little while for more data. Contrariwise, the second student might be urged to make a decision if it appears they are waiting too long for additional information.

However, to make effective heuristics of this nature will probably require the assistance of domain experts and cognitive psychologists. Another interesting problem is that the training content, in addition to teaching style, might need to be altered to take personality factors into account. An example of this would be performing a complex exercise. If a student has a low anxiety threshold it might be useful to have them perform easier problems first to build their confidence before tackling the actual problem, i.e. add additional content based on personality. Finally, it might be the case that other personality indices not normally associated with the learning style literature might be more useful in an AIS. This is yet another reason for including the flexibility to add author defined constructs into the AIS.

Conclusion

This report begins with a discussion of the idea of a learning style, and the assumption that an adaptive instructional system can teach to these styles to improve performance. An accepted way of classifying various competing models is presented, along with summaries of several of these models. Finally, the requirements of a model for use in an AIS, as well as some models that meet these requirements, are presented. This report serves as a starting point for implementing the ability to teach to an individual's learning style in an AIS. This paper also demonstrates the need to implement the AIS in such a way as to allow the instructor to add new learning style models in the authoring tool and to allow for domain specific style/personality interactions with training content.

References

Armor Captains Career Course (2000). *Frustrations and Learning Styles* [On-line]. Available: http://147.238.160.41/

Canfield, A. (1980). *Learning styles inventory maual.* Ann Arbor, MI: Humanics Media.

Center for New Discoveries in Learning (2000). *Personal Learning Style Inventory* [On-line]. Available: http://www.howtolearn.com/personal.html

Consulting Psychologists Press (2000). [On-line]. Available: http://www.cpp-db.com

Cooper, S. E., & Miller, J. A. (1991). MBTI learning style-teaching style discongruencies. *Educational and Psychological Measurement, 51,* 699-706.

Curry, L. (1987). *Integrating concepts of cognitive learning style: a review with attention to psychometric standards.* Ontario, Canada:Canadian College of Health Service Executives.

Dunn, R. (1984). Learning style: State of the science. *Theory into Practice, 23,* 10-19.

Federico, P. A. (2000). Learning styles and student attitudes toward various aspects of network-based instruction. *Computers in Human Behavior, 16,* 359-379.

Felder, R. M. (1996). Matters of style. *ASEE Prism, 6(4),* 18-23.

Gregorc, A. (1982). *Gregorc style delineator.* Maynard, MA: Gabriel System, Inc.

Hickcox, L. (1995). Learning styles: a survey of adult learning style inventory models. In R. Sims & S. Sims, *The importance of learning styles: understanding the implications for learning, course design, and education* (pp. 25-46). Westport, CT: Greenwood.

Keefe, J. W. & Monk, J. S. (1986). *Learning styles profile examiners' manual.* Reston, VA:National Association of Secondary School Principals.

Kiersey, D. M. (2000). *Keirsey Temperament Frequently Asked Questions* [On-line]. Available: http://www.keirsey.com/faq.html

Kirby, J. R., Moore, P. J., & Schoefield, N. J. (1988). Verbal and visual learning styles. *Contemporary Educational Psychology, 13,* 169-184.

Knowyourtype.com (2000). [On-line]. Available: http://www.knowyourtype.com

Leslie, S., Perry, G., & Landrum, M. G. (2000). Keeping distance learners engaged: Design and support issues. In *Proceedings of ASTD Techknowledge 2000* (pp. 19-28). Alexandria, VA: ASTD.

McLoughlin, C. (1999). The implications of the research literature on learning styles for the design of instructional material. *Australian Journal of Educational Technology, 15,* 222-241.

Murray-Harvey, R. (1994). Conceptual and measurement properties of the productivity environmental preference survey as a measure of learning style. *Educational and Psychological Measurement, 54,* 1002-1012.

Newstead, S. E. (1992). A study of two "quick-and-easy" methods of assessing individual differences in student learning. *British Journal of Educational Psychology, 62,* 299-312.

Noe, R. (1999). *Employee training and development.* Boston, MA: Erwin McGraw-Hill.

O'Brien, L. (1989). Learning styles: Make the student aware. *NASSP Bulletin, 73 (Oct),* 85-89.

O'Conner, T. (2000). Using learning styles to adapt technology for higher education [On-line]. Available: http://web.indstate.edu/ctl/styles/learning.htm

Riding, R. J. (1997). On the nature of cognitive style. *Educational Psychology, 17,* 29-49.

Silver, H., Strong, R., & Perini, M. (1997). Integratig learning styles and multiple intelligences. *Educational Leadership, 55 (Sep),* 22-27.

Smith, D. & Kolb D. (1996). *User's guide for the learning style inventory: a manual for teachers and trainers.* Hillsdale, NJ:Hay/McBer Training Resources Group.

Swanson, L. (1995). *Learning style: a review of the literature.* The Claremont Graduate School.

Willcoxson, L., & Prosser, M. (1996). Kolb's learning style inventory (1985): review and further study of validity and reliability. *British Journal of Educational Psychology, 66,* 247-257.

7.3 C4I ITS Use of AIS

7.3.1 Current C4I ITS Work: FBCB2/Tactical Decision-Making Intelligent Tutoring System for Company Commanders

Project Overview

SHAI is researching for STRICOM, an Intelligent Tutoring System (ITS) for tank and mechanized infantry company commanders and has developed a limited prototype. The ITS teaches tactical decision-making and the proper tactical use of FBCB2, a C4I System, by presenting course material and examples, then testing the commander in tactical situations displayed as FBCB2 overlays or in a commercial tank game simulator interfaced to the actual FBCB2 software and the ITS. The ITS monitors student actions in the simulated scenario, assesses their correctness in the current situation, and debriefs the student by automatically assembling an After Action Review (AAR). It then infers the knowledge deficiencies of the student, and formulates a remedial instruction plan, which normally includes further course material, examples, and further exercises to practice and test the student's weaknesses. The eventual intent is to embed the ITS with the FBCB2 software on the various weapon platforms including the M1 Abrams and M2/3 Bradley.

Army C4I System tactical operations and decision-making are complex cognitive tasks that normally require the availability of an instructor. This prevents the effective use of embedded systems for training out in the field, where an instructor is not typically present. Our ITS will assume the duties normally performed by the instructor. In this ten-month effort we have investigated the interfacing requirements, and determined the student and instructor user requirements. We have knowledge engineered instructor and FBCB2 experts, designed the ITS, implemented a limited ITS prototype, and interfaced the ITS prototype to FBCB2 and the commercial simulation game.

Executive Summary

Ultimate Project Goals:

- Improve C4I Readiness
- Provide C4I Instruction to support Embedded training, anytime, anywhere
- Provide more practice to trainees, with feedback and remediation
- Improve Tactical Decision-making by providing more tactical decision-making practice
- Investigate the feasibility of an ITS to support C4I Training
- Implement a limited prototype, interfaced to a C4I system, to prove feasibility

Findings:

- FBCB2 Training decays quickly
    - A very complex system designed for a diverse set of users
    - Most user types will use only a small fraction of the FBCB2 Functionality
        - Armor Company Captain will create FragOs and overlays before combat using 2/3 dozen of the several thousands of available symbols
        - Captain will monitor friendly and enemy positions during combat
        - Captain will issue SITREP after combat
        - Captain will not usually send any messages during combat
    - An embedded, scenario-based training aid would increase combat readiness substantially

- Company Commanders would benefit from more tactical decision-making practice
  - 2-D Map, Tactical Decision Games (TDGs)
  - 2-D Map Dynamic Simulations
  - 3-D Virtual Terrain, Dynamic Tactical Simulations
- Easy to use scenario authoring tools are needed, to enter many scenarios

Limited ITS Prototype built in SHAI's Internet ITS Authoring Tool (IITSAT):

- 3 Parts – FBCB2 Use, 2-D Tactical Decision-Making, 3-D Dynamic Scenarios
- System automatically picks the best scenario for exercises for the particular student
- System automatically debriefs student performance - correct/incorrect/omitted actions and decisions, and the associated underlying principles needed for correct decision-making
- Maintains Model of what the student knows and can APPLY in operational, tactical situations
- Different Instructional Methods for Different Students

Demonstration Sequence:

- First Student, no experience (One Instructional Method (IM), No skipping)
  - Sees detailed FBCB2 CourseWare and Symbol Placement Scenarios
  - Sees Detailed Tactical CourseWare before getting related exercises
  - Mistakes in Tactical Decision Games (TDG) with feedback
- Second Student, has good experience (Different IM/skipping/progressing/modeling)
  - Skips FBCB2 Tutoring/Scenarios
  - Gets TDGs Right and Progresses Out of the TDGs / General Tactical Principles
  - Gets Spearhead Scenarios Wrong, Gets Automatic Debriefs, and sent back for remediation on specific principles (e.g., Fix and Flank, Bounding Overwatch)

ITS Prototype Description

When a new student logs on he is first asked some questions about his background, experience, and last FBCB2 training/use. These questions include level of education achieved, rank, highest unit commanded, types of units served in, computer familiarity/comfort, FBCB2 familiarity/comfort, and general perceptions as to its usefulness. The ITS uses this information to make initial estimates as to the student's mastery of various principles, including both tactical knowledge and the use of FBCB2. It will also be used to select scenarios, other exercises, types of hints, and other forms of instruction. Mastery categories are Beginner, Novice, Intermediate, and Expert. The Beginner category for a principle occurs when a student performs successfully with it less than 20% of the time. (Novice – 20 to 50%, Intermediate – 50 to75%, Expert > 75%.) Students at the expert or intermediate level for a principle are never given hints.

If the ITS estimates that the student's mastery of FBCB2 principles is low, then before doing simulated exercises, the student will first be put through FBCB2-only refresher exercises. An introductory lesson will explain with detailed steps how to create an overlay and find and place the most relevant symbols.

After the FBCB2 refresher exercises (if they were needed), the ITS will begin tutoring the student on general tactical principles. If it estimates his mastery is relatively high it will proceed immediately to tactical decision games presented and answered as FBCB2 overlays. If not, it will first present General Tactical Principle Courseware. For each Tactical Decision Game (TDG), the ITS will analyze the student's plan (given as an FBCB2 overlay) and automatically create a debriefing describing what parts of his plan are right, what parts are wrong, and giving an expert's rationale for the best options. For poor decisions, the ITS will lower its estimate of the mastery of principles related to those decisions, and provide remedial materials on those principles, before presenting anymore TDGs. The student's overlay is evaluated to comparing to overlays input by an instructor for that particular TDG. These typically represent a few possible right answers and a few common mistakes. The instructor will also have annotated the overlays with information for use by the ITS in assembling the debrief and determining which principles the student is weak in.

For the TDGs and the 3-D scenarios, the ITS will initially select exercises based on the need to test untested principles, following each by a debriefing and detailed information on the principles missed. The ITS will then begin to also retrieve scenarios that exercise the principles in which the student's mastery is weakest. Furthermore, for any scenario using principles that the ITS believes the student is weak in, it will provide him hints for the scenario, if they are available. These are generally questions designed to get him to think about the most important tactical principles required in the scenario.

After the student has demonstrated (or learned) his mastery of general tactical principles in the TDGs, he proceeds to that portion of the course that requires him to show that he can apply these same principles in a 3-D virtual reality dynamic tactical simulation. Additionally, more operations-oriented principles (such as knowing when and how to use a company wedge formation) will also be tested. In the prototype, the student is given a short situation description and then proceeds to execute the mission in Spearhead II. After the scenario ends, the event log is analyzed by the ITS to automatically determine which actions were correct, incorrect, or omitted, and the underlying principles that were understood and applied or not.

In some scenarios, we will have subordinates that do not follow orders, plans, and proper tactics. Normally the commander would correct these problems with voice commands. For this demonstration we will do no Speech Understanding. But these corrections should be manifested by the motions and actions of the commanders' company's tanks. The ITS will assess these motions and actions (captured from VMF messages). For example, the commander's OPORD may have had the lead platoon in a wedge formation but it is proceeding in a column. If he orders them into the correct formation, an evaluation machine will detect the correction and he will get credit for recognizing the wrong formation, and recognizing the need to correct it. If they continue to move as a column, he will fail these principles.

Some scenarios will in particular test his use of FBCB2 to maintain situational awareness. For example, we can have enemy approaching from an unexpected direction, which is trivial if the commander is watching the FBCB2 map display. Another test is to have friendlies show up suddenly at an expected enemy location. A test for the combat principle of audacity is to have the commander unexpectedly come across a much larger force in a totally unprepared situation, such as refueling, without security.

In the scenario, unplanned actions will occur, such as unexpected contact with the enemy. His tanks will begin to react and he will also issue particular orders, verbally in the real world, with mouse clicks in the simulation. Again, the correctness of his decisions can be evaluated from the movements and actions of his company's tanks. For example, one scenario involves the lead platoon spotting a roadblock at a choke point. That platoon should deploy in a support by fire position and the commander should order his infantry to protect each flank. He should then order dismounted assaults up each flank and around the roadblock to secure the far side. He should Call For Fire at appropriate locations and times during the scenario as well. Evaluation machines check each of these actions and debrief him at the end as well as infer the state of his tactical knowledge.

The student is also evaluated based on the motions and actions of his individual tank. These correspond in many ways to how individual tank commanders would be evaluated, but also include additional factors, such as not overly endangering the company commander, that the commander should be with the main effort, and that when traveling with a platoon in the wedge formation, the commander's tank should be at it's center.

After the scenario, the commander is debriefed with an After Action Review. All the things he did right and wrong are reviewed and he is told about the relevant principles. For the failed principles he is given detailed information and one example for each. The mastery level estimates for all principles involved are updated. Based on these, a new scenario is retrieved. Scenarios are selected that test untested principles and test recently failed principles.

The prototype has different instructional methods for students with little mastery or experience compared to students with a lot of mastery and experience.

7.3.2 Additional Adaptive Capabilities Needed in the C4I-ITS

Personality and other Individual Difference Issues

FBCB2 is a advanced, highly complex software system being introduced into vehicles manned by personnel that heretofore had no need for the skills required to successfully utilize complex software systems. Thus, many of the Soldiers who will need to be trained on FBCB2 have little or no experience with any software, much less software of such complexity. In addition to experience differences, there are deeper personality issues. Some Soldiers, while bravely facing enemy fire, are actually scared of the prospect of using software. Others have a mistrust of software. And still others doubt the wisdom or usefulness of such software in their combat vehicles. A more general trait that varies among Soldiers is how quickly they can learn new concepts and ideas, especially those that are very different from any they have learned before. A related problem, for those with E-mail and other electronic system experience is learning in what ways FBCB2 behaves differently than non-tactical systems they have used that appear very similar. Given the enormous size, complexity, and number of features in FBCB2, some Soldiers will have more of an ability to concentrate on just a very small fraction of capabilities that they need, while filtering out the enormous clutter presented by the sheer number of features that they should ignore for their own particular job.

Proper tactical decision-making is greatly affected by the personality of the decision-maker in a large numbers of ways. In fact, some of the general tactical principles are themselves personality traits. The best example is boldness. That is, a good tactical plan should often be bold. This requires a certain boldness in the person creating the plan. Similarly, a good tactical decision-maker can make decisions, on his own initiative. Many tactical decisions have to made fast, so the speed with which the Soldier can process information and consider tactical principles is important. It is much more important to follow the commander's intent than his specific orders, since a changing situation can make the original orders obsolete. This will be easier for different personality types than others. Similarly as described in earlier sections, some students tend to concentrate on details while others like to see the bigger picture. But understanding the bigger picture is an important tactical principle. Similarly, different types of students will find it more difficult to understand the enemy, his intent, and the fact that he is a thinking, acting entity with his own motivations and goals and he will not only react to our actions but will try to seize the initiative himself.

Learning Style Issues

The C4I ITS is implemented with the Internet ITS Authoring Tool (IITSAT). This is fortunate since IITSAT allows the representation and use of alternate instructional methods. For example, one instructional method is to first present an overview, followed by examples, followed by more detailed information and then exercises. Another method is to jump immediately to exercises. Thus, if different learning styles are identified in the students in the C4I domain, these can readily be taken advantage of, given that the AIS capabilities will be incorporated into IITSAT.

Some learning style differences (and therefore different instructional methods) are already apparent. Students who are anxious about using software and FBCB2 will have to be presented the material at a much slower pace to keep them from having severe emotional reactions. Students familiar with software in general can be taught by having the system point out the similarities and differences between FBCB2 and common software packages.

7.3.3 Phase II Work

As mentioned above, AIS capabilities will be incorporated into IITSAT during Phase II. This will allow C4I to readily take advantage of the AIS capabilities. The current plan for C4I ITS is to increase the capabilities of the prototype and begin Soldier trials. This work will occur in parallel with the AIS Phase II which will facilitate synergy between the two projects.

Learning Styles

The AIS version of the C4I ITS will identify the student's predominant learning style and select instruction accordingly. Currently, many of the general tactical principles have multiple descriptions. These can be in different media (text, speech, graphical, and animations) and of different themes (quotes from historic figures, descriptions highly correlated to scenarios (exercises or examples), general first principle discussions, historic scenarios, small examples, etc.). Currently, which description is presented to which student is fairly arbitrary. AIS capabilities will allow the C4I ITS to at least select the first mode and content based on the student's preferred learning style. For example, if the student is classified as preferring a reading mode, the system would tend to present a text description of a principle. Similarly, if the student's cognitive style is case-based, the historic example-based description of the principle might be retrieved. If the student tends to reason from first principles, a description of the principle in terms of first principles can be given. For example, a description of the fix and flank principle can either refer to historic cases where that tactical principle was aptly demonstrated, or, could describe it more in terms of the fixing force maintaining the attention and direction of the enemy's weapon systems, while the flanking maneuver allows an attack from an unexpected direction (surprise) and from a direction toward which the enemy's weapon systems are not directed.

Students who have been classified as active experimentation learners would have the tactical exercises emphasized to them. Those classified as reflective observation learners would have a greater emphasis on descriptions and examples, though exercises would still be used to test the student's mastery.

Personality and other Individual Difference Issues

In the case of learning FBCB2, the most important personality and other student differences (besides the mastery differences already tracked) are fear of software, current anxiety level, mistrust of software, bias against use of software in a combat vehicle, and openness to new concepts. For FBCB2, informing the student of his personality factors (he will already know them) will not be useful and predicting his specific actions based on personality factors to more proactively guide him is not practical. This leaves adapting the instruction and content to the student's personality factors. As mentioned above, students with a fear of software or a high current anxiety level, must be presented information at a slow pace, and given lots of small simple examples to cement the material. Their mastery of the individual concepts and techniques must be estimated high with a high degree of certainty before proceeding to more complex exercises requiring combinations of principles. Those with a mistrust of software, a bias against FBCB2, or a lack of openness need early exercises where the benefits are clearly demonstrated. An ambush and/or fratricide-oriented scenario exercise illustrates the combat benefits of FBCB2 in very concrete terms. Similarly, example operational scenarios with and without FBCB2 overlay capabilities available demonstrate the usefulness of FBCB2 before combat.

As mentioned earlier, proper tactical decision-making is greatly affected by the personality of the decision-maker. In addition to being able to use personality factors to adapt instruction, it is also practical and useful to inform the student about them and to predict types of actions in order to provide hints to proactively prevent the student from making mistakes based on those personality factors. Consider boldness for example. If we find this trait lacking in a student, we can have specific remedial instruction, illustrating the historic importance of boldness in military planning. Furthermore, we can tend to retrieve exercises for the student where boldness is an important factor, thus practicing his weakness. But given a specific tactical scenario where there are two courses of action as the bolder one is more correct, using MAMID or similar techniques, the ITS can predict that the student will likely choose the less bold, suboptimal option. Before the decision is made, just encouraging him to "Be Bold" would be extremely helpful to him. Finally, keeping the student informed of his tendency to not be bold enough would help him to intellectually compensate during tactical planning. Similarly techniques can be applied to the other personality factors with tactical decision-making ramifications.

8. Methodologies

8.1 Intelligent Tutoring Systems

ITSs contrast with most Computer Based Training (CBT) systems in that the latter can usually be described as automated text books. That is, most CBTs are developed by using the same approach as a corresponding textbook. In some domains, multimedia material that textbooks cannot include, such as video, audio, and animation, are added, but these don't really reflect differences in instructional methods. Other than allowing self-navigation, typical CBTs do not attempt to adapt or tailor the instruction to the individual. Additionally, most CBTs do not embody any particular instructional approach, theory, or philosophy, other than the instructional approach which happened to exist in the textbook on which the CBT system is based.

ITSs, on the other hand, emphasize custom instruction tailored to the particular individual and are typically based on pedagogical concepts. To truly tailor instruction requires that the instruction system create, develop, and maintain a model of the student, which ITSs do and most CBTs typically do not. This model is used as a basis for instruction method and content selection, diagnosis, remedial course formulation, re-testing, and progress monitoring and reporting, all done automatically.

8.2 Decision-Centered Design and Cognitive Task Analysis

Too often, decision support and intelligent systems and aids are not designed around the actual decision requirements of the task. As a result, the systems often fail to provide the necessary information, fail to provide it in a useful form, or, as is often the case, make it more difficult to access essential information. In the field of human factors (Woods, Johanessen, Cook, & Sarter, 1994), this is known as clumsy automation, because the good intentions of the designers result in worse performance, rather than improved performance.

The worlds of intelligent tutoring and aiding offer perfect opportunities to attempt to address some of these design issues and problems. A key aspect of accomplishing this involves applying the recently developed methodology of Decision-Centered Design (Klein, Kaempf, Wolf, Thordsen, & Miller, 1997). This approach begins with a Cognitive Task Analysis (CTA) to identify the decision requirements of the task: the key judgments and decisions, the reasons why they are difficult, the types of errors that are found, and the patterns and strategies used by experienced personnel. What is striking about this approach is that it does not begin with decompositions of a task into basic elements, or determinations of information flow. Instead, it fits within user-centered design approaches (e.g., Landauer, 1995) by focusing on the decision requirements for performing the task well, and uses these to design the architecture of the system. Klein Associates has successfully applied this method in previous design projects (Klinger & Gomes, 1993; Miller & Lim, 1993) with high degrees of success. Klinger, Andriole, Militello, Adelman, Klein, and Gomes (1993) report a careful evaluation that determined that performance of AWACS Weapons Directors was significantly improved by an interface designed according to decision requirements. Miller and Lim (1993) designed a decision support system for Air Force weaponeers, and its value was so obvious that the sponsors moved directly into system development. More recently, DCD has been successfully applied in proposing redesigns for the platform displays for Naval landing signal officers (LSOs) (Stottler & Thordsen, 1997; Thordsen, 1998), developing distributed team training models and training development tools for naval air missions (Thordsen et al., 1998) and is currently being applied in projects examining the overall team coordination/performance aspects of the AWACS and naval distributed team training. Accordingly, we expect that the opportunity to use a Decision-Centered Design (DCD) approach in the current effort will result in a foundation of the intelligent tutoring system that will have direct appeal and value for individuals involved both in the giving and receiving of instruction.

8.3 Case-Based Reasoning

SHAI plans to employ case-based reasoning as a key reasoning method used by the AIS, to assess student performance and other attributes, and make instructional decisions such as selecting and configuring practice and diagnostic lessons.

Case-Based Reasoning (CBR) is a field of Artificial Intelligence which deals with the method of solving a current problem by retrieving the solution to a previous similar problem and altering that solution to meet the current needs. Case-Based Reasoning is a knowledge representation and control methodology based upon previous experiences and patterns of previous experiences. These previous experiences, or "cases" of domain-specific knowledge and action, are used in comparison with new situations or problems. These past methods of solution provide expertise for use in new situations or problems. From our previous ITS experience, we believe that the general problem of teaching students is well suited for the application of a Case-Based Reasoning method.

CBR systems offer enormous benefits compared to standard AI approaches. The knowledge elicitation bottleneck is largely circumvented. Cases can be automatically acquired directly from domain experts. Rules, on the other hand, almost always require the intervention of a knowledge engineer. Instead of having to elicit all of the knowledge required to derive a solution from scratch, only the knowledge required to represent a solution is needed. So knowledge elicitation is largely avoided with CBR and may be completely automated depending on the type of application and the expert. This makes CBR especially appealing for an instructional design framework that will potentially be applied to multiple domains because it reduces the knowledge engineering time requirement.

Conventional knowledge base technology dictates a single, fixed problem-solving methodology. With CBR, each case (in the extreme), can represent a different methodology. This is important for complex domains where different problems or situations, although sharing the same fundamental concepts, may require different solution strategies.

We plan to use case-based reasoning to select and modify instructional plans. For example, a case could represent a lesson, task, or scenario that achieves certain instructional goals. The Instructional Planner will select the case and modify it to achieve additional instructional goals.

9. Related Work

9.1 Relevant SHAI Projects

9.1.1 AEGIS Tactical Action Officer ITS

SHAI has developed for the U.S. Navy a simulation-based adaptive tutoring System (also known as an intelligent tutoring System (ITS)) which enables students to act as TAOs in tactical simulations. The simulation's graphical user interface displays a geographical map of the region and provides rapid access to sensor, weapon, and communication functions. After the student completes a scenario, the ITS evaluates the entire sequence of student actions to infer tactical principles that the student correctly applied or failed to apply. These principles are detected according to sophisticated pattern-matching algorithms defined by the instructor using the System's graphical user interface. The System is highly configurable within the domain of naval tactical simulations, and authoring tools enable the instructor to define new types of ships and aircraft, scenarios, and principles. The instructor can also define complex behaviors for each friendly and enemy ship and aircraft to create realistic, multi-agent simulations. The TAO ITS has proven an effective training tool. It is currently being used by tactical action officer students at the Surface Warfare Officers School. Initial independent evaluation of the software in use has been highly favorable. Simulation-based intelligent training systems complement traditional classroom or computer-based training by enabling students to practice the application of concepts and principles. Additional funding has been received to adapt the TAO ITS for fleet use on board ships. Contact: Joe Russell (703) 602-5959 x183.

9.1.2 C4I ITS

Under contract to STRICOM, SHAI is currently developing C4I ITS, a prototype intelligent tutoring system for armored and mechanized infantry company commanders. C4I ITS will teach tactical decision-making, command and control principles, and the use of the Force XXI Battle Command Brigade and Below (FBCB2) command and control system. Before each mission, students will issue pre-mission orders and graphical overlays that specify movements. The tutoring system will assess these plans using symbolic pattern recognition techniques that compare each student's plan with annotated good and bad plans (or portions of plans) supplied by experts. Similarities between the student's plan and the good plans will identify specific proficiencies in high-level and low-level skills. Similarities between the student's plan and bad plans will identify skill deficiencies.

During the mission, the students will interact with FBCB2 and Spearhead, a military game simulation program that is being adapted for use in military training and is being integrated with FBCB2. Spearhead implements the simulation behaviors and presents a 3 dimensional "out the window" view of the world as seen from each vehicle. Spearhead will exchange real-time simulation data with FBCB2, and C4I ITS will intercept those packets. The tutoring system will construct scenarios and modify scenarios in progress to include situations that test various situation assessment skills. For example, the tutoring system could place friendly forces in certain locations that the student might otherwise fire upon. If the student does fire upon those positions, the tutoring system can infer that the student failed to use FBCB2 to notice the presence of friendly forces in that area. As another example, the tutoring system could place enemy forces at a location such that the student should direct friend forces to oppose them. Failure to perform this action indicates either a lack of situation awareness of the enemy forces, or a poor tactical decision. Contact: Rodney Long at STRICOM: (407) 384-3928.

9.1.3 Internet Intelligent Tutoring System

SHAI is constructing for the U.S. Air Force an Intelligent Tutoring System (ITS) server that connects, and promotes communications between, a loose confederation of ITSs maintained by individuals with little or no knowledge of each other's existence over the Internet. A user interacts with one ITS. When it determines that the student lacks knowledge in a related field which is handled by another ITS, it sends him or her there.

A key goal of this project is the development of the Internet Intelligent Tutoring System Authoring Tool (IITSAT) that enables instructors to specify hierarchically-organized learning objectives and curriculum elements, as well as instructional strategy decisions such as criteria for skill mastery and algorithms for selecting next lessons and review materials. ITSAT is designed to reduce the cost and difficulty of developing effective ITSs, by streamlining the process of specifying these instructional strategies. The Phase II project started in April, 1998 and the contact is Teri Jackson at 210-536-3908.

9.1.4 Intelligent Tutoring System for Adult Literacy Enhancement

25% of the Navy's enlisted population scores below the eighth grade level in literacy skills. There is a need, therefore, to improve the reading ability of adults up to the twelfth grade level. SHAI is developing an Intelligent Tutoring System to improve the literacy skills of adults. The ITS can model a student's reading abilities and provide customized instruction. The ITS also includes an authoring tool that allows non-programmers to expand the set of reading material available to the tutor. The ITS and the authoring tool resulting from Phase II will be of benefit not only to the Navy, but also to other branches of the military and the government as well as to the adult community in general. The ITS can be used at adult education centers, job corps training centers, and by other commercial organizations desiring to improve literacy among their employees. This Phase II project shows our ability to develop ITSs for use by under-performing students. Contact: Dr. Susan Chipman at (703) 696-4318.

9.1.5 Dismounted Infantry Military Operations in Urban Terrain (MOUT) Intelligent Tutoring System

SHAI developed, in cooperation with Research Development Corporation, a Simulation-based ITS (SITS) for training dismounted infantry, both as individuals and teams, that would be used with a virtual reality simulator. Included in the project was the development of a generic ATS architecture that can interface with existing and future simulators. The SITS diagnoses student learning needs, determines what instruction content and technology are most appropriate, and drives the presentation of that instruction. Key technologies are case-based reasoning (CBR), integrated knowledge structures for representing expert and student knowledge, automatic knowledge elicitation, and dynamic scenario selection and creation. The architecture supports automatic and semi-automatic knowledge engineering to update its knowledge base as the domain itself evolves. The System trains squad and fire team leaders in Military Operations in Urban Terrain. The ATS monitors the student's actions in the virtual reality environment, assesses his deficiencies, and modifies the scenario or creates new ones to address those deficiencies. A CBR system also selects the most appropriate instructional technique based on the student's individual requirements and past learning behavior.

9.1.6 Task Tutor Toolkit and Remote Payload Operations Tutor for Procedural Training

To lower the cost and difficulty of creating scenario-based intelligent tutoring systems for procedural task training, Stottler Henke Associates, Inc. (SHAI) worked closely with the Operations Training Group at NASA's Marshall Space Flight Center to develop the Task Tutor Toolkit (T^3), a generic tutoring system shell and scenario authoring tool. The Task Tutor Toolkit employs a case-based reasoning approach where the instructor creates a procedure template that specifies the range of student actions that are "correct" within each scenario. The system enables a non-programmer to specify task knowledge quickly and easily via graphical user interface, using a "demonstrate, generalize, and annotate" paradigm that recognizes the range of possible valid actions and infers general principles that are understood (or misunderstood) by the student when those actions are carried out. The annotated procedure template also enables the Task Tutor Toolkit to provide hints requested by the student during scenarios, such as "What do I do now?" and "Why do I do that?" At the end of each scenario, RPOT displays the principles correctly or incorrectly demonstrated by the student, along with explanations and background information. The Task Tutor Toolkit was designed to be modular and general so that it can be interfaced with a wide range of training simulators and support a variety of training domains.

SHAI and NASA used the Task Tutor Toolkit to create the Remote Payload Operations Tutor (RPOT), a tutoring system application which lets scientists who are new to space mission operations learn to monitor and control their experiments aboard the International Space Station according to NASA payload regulations, guidelines, and procedures. NASA is currently evaluating the effectiveness of RPOT and the Task Tutor Toolkit and is exploring other potential training applications for the Task Tutor Toolkit. Contact: Mr. Stephen Noneman, (256) 544-2048. Phase II Completed: February, 2000.

9.1.7 Operator Assessment and Operator Machine Interface Enhancement (OA/OMIE)

SHAI is developing for the Navy an intelligent, Operator Assessment and Operator Machine Interface Enhancement (OA/OMIE) system for the LAMPS SH-60R Multi Mission Helicopter. During training simulations, the system tests operator knowledge through the use of tactical scenarios and derives the operator's mental model based on his performance and explanations for his actions. The system then adapts the operator's interface, based on deficiencies revealed in the mental model. This adaptation involves the coordination of a collection of decision aids that draw upon a variety of disparate sensor data, as well as mission intelligence and tactical knowledge, to enhance sensor employment, enemy platform classification, situational awareness, and overall probability of satisfying mission objectives. The user-modeling and intelligent interface technology developed in this project is highly applicable to supporting adaptive embedded help. Client: Naval Air Systems Command HQ. Contact: Lt. Commander Henry Jackson, 301.757.8159. Phase I Completed: December, 1997. Phase II ongoing.

9.1.8 Constructivist Distance Learning System for Counter-Terrorist Intelligent Analysis

SHAI is developing for the U.S. Army at Ft. Huachuca a training system comprised of two parts. The first is a general framework that supports the creation of Constructivist DL courseware in a wide variety of areas. The second product is a specific tutoring and scenario authoring system. The Intelligence in Combating Terrorism (ICT) courseware was built using this general course creation framework. This tutoring system gives the students extensive, hands-on training in the analysis of raw intelligence information related to investigating terrorist organizations and installation threat assessments.

The purpose of the ICT tutor is to train the student in the analysis of raw intelligence leading to a compact summary of a terrorist operation and an assessment of the current level of threat. The tutor uses

Constructivist learning theory by supporting adaptive learning, modeling, intentional activity, and rich scenario contexts. The tutor immediately places the learner in a 'real-world' environment that allows the students to learn in context and apply what they have learned. It is this contextual experience of knowledge acquisition in an authentic environment that facilitates the learner to create his own constructs that can be applied to new, unfamiliar situations.

The Tutor presents the student with a problem, provides the student with the tools to solve the problem, and offers customized suggestions as a resource to solve the problem. The student produces a solution, receives consequences based on their solution, and is guided to suggested areas for review before beginning another scenario. Contact: Helen Remily at (520) 533-9077.

9.1.9 Intelligent Tutoring System for Long-Range Acoustic Detection of Submarines

SHAI is developing for the U.S. Navy an *Acoustic Analysis Intelligent Tutoring System* (AAITS) which will enable students to practice the detection and classification of sources of underwater acoustic signals such as submarines and whales. Acoustic analysis experts will create scenarios using a Scenario Authoring Tool by selecting and viewing LOFARGRAMs which are frequency-analyzed acoustic datasets displayed as 2D images, annotating them with significant features and links among related features, providing reasons for requesting each LOFARGRAM, and assigning a final classification. Students will use the Tutoring System to carry out this same acoustic analysis. By comparing the details of each student's analysis with those of the expert, the Tutoring System can identify the acoustic analysis principles understood and correctly applied by each student, provide specific and individualized feedback, suggest relevant training materials, and select appropriate next scenarios. By storing LOFARGRAMs annotated by experts, AAITS also serves as a knowledge repository which disseminates the most current acoustic analysis expertise to sonar technicians on land or at sea. A key innovation of AAITS is the use of an application-specific Scenario Authoring Tool which enables experts to create scenarios which encode their expertise and analyses intuitively, by annotating datasets. Phase II project start date: Feb, 1999. Contact: Master Chief Joseph Spivey at SPAWAR, (858) 537-0312.

9.2 Related Work by Others

Psychology research in the effect of individual differences on skill acquisition dates from the turn of the century. However, much of this research focused on predicting the student's skill acquisition rather than on determining how to adapt instruction, based on individual student attributes, to maximize learning for each student. Although research into the utility of adapting computer-based training to individuals is nearly as old as the field of computer-based training, much of this research has focused on interactions between general aptitude level and learning environment, rather than on the full range of other student attributes. For example, Campbell (1964) found that high aptitude students fared better using "self-direction" learning environments whereas low aptitude students fared better using less flexible "programmed instruction" environments. This *aptitude-treatment interaction (ATI)* was confirmed by Shute and Glasser (1990) in a study where 800 students used Smithtown, a discovery learning environment for learning microeconomics principles. Results from this study showed that some subjects learned very effectively and rapidly in this exploratory environment (in half the time typical of classroom instruction), whereas other subjects did not. Cronback and Snow (1977) reported that high-aptitude subjects learn better when they can control the learning environment to process information in their own way, whereas low-aptitude subjects tend to do worse when provided this control.

In another experiment reported by Shute (1992) involving 282 subjects, an automated tutoring system taught simple electronics circuit analysis using two modes of feedback. In *rule-induction* feedback mode, the tutoring system provided feedback that identified the relevant variables in the problem, but the student had to induce the relationships among the variables. In *rule-application* feedback mode, the feedback explicitly stated the variables and their relationships for a given problem. Students then applied the rule to the problem as directed by the feedback. This experiment showed significant *interactions* between the student's cognitive aptitude (as measured by verbal, quantitative, and spatial associative learning skills), the feedback mode presented to the student, and the student's performance in four post tests that measured declarative knowledge acquisition and procedural skill acquisition. Specifically, low ability subjects learned declarative skills more effectively when taught in rule-application mode and performed poorly in procedural tasks regardless of feedback mode. By contrast, high ability students learned declarative skills more effectively when taught in rule-induction mode, and they learned procedural skills more effectively when taught in rule-application mode.

Hudlicka (1999) describes a Methodology for Analysis and Modeling of Individual Differences (MAMID), which provides a generic method for representing a variety of individual differences factors in human performance models. The proposed methodology consists of four steps: identification of cognitive processes and structures mediating performance; design of a corresponding parameterized model; identification of cognitive, affective, and personality factors affecting model processes; and encoding of identified factors in terms of model parameters and knowledge bases. The MAMID methodology is being implemented within an integrated simulation environment, providing a testbed for analysis of distinct individual profile effects on task performance. The demonstration scenario involves an Army COA execution task, which is particularly susceptible to individual differences variations. MAMID is applicable to both individual and team settings, and can be incorporated within a variety of agent architectures.

10. References

Ackerman, P. L. (1987). Individual differences in skill learning: An integration of psychometric and information processing perspectives. *Psychological Bulletin*, 102, 3-27.

Adam, J. A. (1987). Historical review and appraisal of research on the learning, retention, and transfer of human motor skills. *Psychological Bulletin*, 102, 41-74.

Anderson, J. R. (1983) *The architecture of cognition*. Cambridge, MA: Harvard University Press.

Campbell, V. N. (1964) Self-direction and programmed instruction for five different types of learning Objectives. *Psychology in the Schools*, *1*, 348-359.

Catell, R. B. (1971). *Abilities: Their structure, growth, and action.* Boston: Houghton Mifflin.

Coyle, S. (1996). The art and science of flying helicopters. Iowa State University Press.

Cronbach, L. J., & Snow, R. E. (1977). *Aptitudes and instructional methods: A handbook for research on interactions.* New York: Irvington.

Dohme, Jack, (1995), *The military quest for flight training effectiveness*, In: W.E. Larsen, R.J. Randle Jr., and L.N. Popish (Eds.), Vertical Flight Training, NASA Reference Publication 1373.

Eysenck, H. (1986). The theory of intelligence and the psychophysiology of cognition. In R.J. Sternberg (Ed.), Advances in the psychology of human intelligence (Vol. 3, pp. 1-34). Hillsdale, NJ: Erlbaum.

Gardner, H. (1983). Frames of mind: The theory of multiple intelligences. New York: Basic Books.

Herrnstein., R. J., & Murray, C. (1994). *The bell curve: Intelligence and class structure in American life.* New York: Free Press.

Hudlicka, E. (1994) *Toward a Computational Model of Attention.* BBN Report. No. 7791.

Hudlicka, E., & Billingsley, J. (1999) Representing Behavior Moderators in Military Human Performance Models. *Proceedings of the 8th Conference on Computer Generated Forces and Behavioral Representation*, Orlando, FL.

Jonassen, D.H., Peck, K.L. & Wilson, B. G. (1999). *Learning with technology: A Constructivist perspective*. New York, NY: MacMillan.

Masie, Elliot (1997), *The Computer Training Handbook: Strategies for Helping People to Learn Technology,* Lakewood Publications.

Montgomery, S., Addressing diverse learning styles through the use of multimedia, *Proceedings of the ASEE/IEEE Frontiers of Education Conference.*

Schank, R. (1977) *Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures,* Hillsdale, NJ: Lawrence Erlbaum Associates.

Schank, R. (1995), *What We Learn When We Learn by Doing*, Technical Report no. 60, Institute of Learning Sciences, Illinois.

Shute, V. J., & Glaser, R. (1990). A Large-Scale Evaluation of an Intelligent Discovery World: Smithtown. *Interactive Learning Environments*, I(1), 51-77.

Shute, V. J. (1992). Aptitude-treatment interactions and cognitive skill diagnosis. *Cognitive Approaches to Automated Instruction*, Hillsdale, NJ: Lawrence Erlbaum Associates.

Stottler, R.H. and Ramachandran, Sowmya (1999), *A case-based reasoning approach to intelligent tutoring systems (ITS) and ITS authoring*, FLAIRS special tracks on Intelligent Tutoring Systems, Orlando, FL.

U.S. Army Aviation Center (1999) Flight Training Guide - Initial Entry Rotary Wing (IERW) Aviator TH-67 Primary Phase.

U.S. Dept. of Transportation Federal Aviation Administration, *Aviation Instructor's Handbook.*

Appendix A: Demo Scripts

To run the Helicopter tutor:

1. Insert the Microsoft Flight Simulator 2000 CD number 2 into the CD-ROM drive and invoke Microsoft Flight Simulator. Press the "Fly Now!" button.
2. Select the Aircraft/Select Aircraft menu choice to select the Bell 206B JetRanger helicopter. Select the Flights/Save Flight menu choice to save this flight and make this aircraft your default.
3. Run C:\HeloTutor\HeloTutor.exe.
4. When the Tutor displays the login window, select a student name from the list box and press "OK".
5. Merlin, the speech-to-text agent, will verbalize the Tutor's reasoning as it selects and configures a lesson. The agent`will then deliver the lesson briefing. You can minimize the HeloTutor application window.
6. In the Microsoft Flight Simulator 2000 menu, select the Flights/Multiplayer-Connect menu choice. A "multiplayer connect" window will popup. Select "TCP/IP" in the Protocol list. Select "Lesson-007" in the Sessions list, and click the "Join" button. This action establishes a connection between the MS Flight Simulator and the Helicopter Tutor program.
7. All the flight parameters will be displayed in a list in the HELOTUTOR window. If you want to save all these data, check "Log Data" in the HELOTUTOR window before running the simulator.
8. Press P to resume the simulator and start the lesson.
9. To exit the program, first disconnect Microsoft Flight Simulator 2000 from it by selecting Flights-Multiplayer-Disconnect in the menu, close the Microsoft Flight Simulator 2000, then click the "x" button on the right top corner of the HELOTUTOR window.
10. To log in as another student, select the Flights/Multiplayer-Disconnect menu choice, exit from the Helicopter Tutor application, re-invoke the Helicopter Tutor application, log-in as another student, and select the MS Flight Simulator Flights/Multiplayer-Connect menu choice to reconnect to the Helicopter Tutor.

An example student model file is shown below. In an actual system, this student model would be read from a file or database at the beginning of each session, updated during the session, and saved to the file or database at the end of the session.

```
(deffacts student-bill
  (skill-level (skill hover)                (automaticity NIL) (proficiency low))
  (skill-level (skill control-heading)      (automaticity NIL) (proficiency medium))
  (skill-level (skill control-xy-position) (automaticity NIL) (proficiency medium))
  (skill-level (skill control-altitude)     (automaticity high) (proficiency high))
  (skill-level (skill perceive-heading)      (automaticity high) (proficiency high))
  (skill-level (skill perceive-xy-position)  (automaticity high) (proficiency high))
  (skill-level (skill perceive-altitude)     (automaticity high) (proficiency high))
  (skill-level (skill straight-level)       (automaticity NIL) (proficiency low))
  (skill-level (skill control-direction)    (automaticity NIL) (proficiency medium))
  (skill-level (skill control-speed)        (automaticity high) (proficiency high))
  (student-profile
    (student-name Bill)
    (student-summary "Bill is in his tenth week of training. His proficiency in hover
flight is less than nominal.
His proficiency in subskills control-altitude and control XY position are only
medium.")
    (preferred-speaking-speed 160)
    (num-weeks-in-training 10))
)
```

After the student model file is selected, the program selects and configures an appropriate lesson, based upon the proficiencies and automaticities estimated for each skill, the number of weeks of training acquired by the student, and other attributes. The reasoning process by which the tutor selects and configures lessons for each student is shown in the logs below.

A.1 Student Bill

Bill is in his tenth week of training. His proficiency in hover flight is less than nominal. His proficiency in subskills control-altitude and control XY position are only medium. Proficiency of skill control-heading is medium and the nominal proficiency for students in week 10 of training is high.

Lesson selection - log of the AIS's reasoning

1. Considering lesson straight-level to increase proficiency in skill control-heading.
2. Proficiency of skill control-xy-position is medium and the nominal proficiency for students in week 10 of training is high.
3. Considering lesson straight-level to increase proficiency in skill control-xy-position.
4. Proficiency of skill hover is low and the nominal proficiency for students in week 10 of training is medium.
5. Creating a goal of increasing proficiency in sub-skill control-heading of skill hover.
6. Considering lesson straight-level to increase proficiency in skill control-heading.
7. Creating a goal of increasing proficiency in sub-skill control-xy-position of skill hover.
8. Considering lesson straight-level to increase proficiency in skill control-xy-position.
9. Considering lesson hover to increase proficiency in skill hover.
10. Selecting best lesson: straight-level. Objective is to learn skill control-heading. Importance = 3. .

Lesson configuration - log of the AIS's reasoning

11. Proficiency in controlling altitude is high. Setting altitude tolerance to 20.0 feet.
12. Proficiency in controlling speed is high. Setting speed tolerance to 5.0 knots per hour.
13. Proficiency in controlling direction is medium. Setting direction tolerance to 10.0 degrees.

Lesson briefing delivered to the student

Bill, please perform straight and level flight. Take off and reach altitude of 2000.0 plus or minus 20.0 feet. Maintain direction of 180.0 plus or minus 10.0 degrees. Accelerate to 50.0 plus or minus 5.0 knots per hour.

A.2 Student Bob

Bob is in his third week of training. His proficiency in all skills is low. He prefers to be spoken to slowly. Proficiency of skill control-heading is low and the nominal proficiency for students in week 3 of training is medium

Lesson selection - log of the AIS's reasoning

1. Considering lesson straight-level to increase proficiency in skill control-heading.
2. Proficiency of skill control-xy-position is low and the nominal proficiency for students in week 3 of training is medium.
3. Considering lesson straight-level to increase proficiency in skill control-xy-position.
4. Selecting best lesson: straight-level. Objective is to learn skill control-heading. Importance = 1.

Lesson configuration - log of the AIS's reasoning

5. Proficiency in controlling altitude is low. Setting altitude tolerance to 100.0 feet.

6. Proficiency in controlling speed is low. Setting speed tolerance to 20.0 knots per hour.
7. Proficiency in controlling direction is low. Setting direction tolerance to 20.0 degrees.

Lesson briefing delivered to the student

Bob, please perform straight and level flight. Take off and reach altitude of 2000.0 plus or minus 100.0 feet. Maintain direction of 180.0 plus or minus 20.0 degrees. Accelerate to 50.0 plus or minus 20.0 knots per hour.

A.3 Student Dave

Dave is in his tenth week of training. His proficiency in hover flight is less than nominal. His proficiency and automaticity in subskills of hover flight are generally high except for the subskill of controlling x y position, whose automaticity is unknown. Proficiency of skill hover is low and the nominal proficiency for students in week 10 of training is medium

Lesson selection - log of the AIS's reasoning

1. Considering lesson straight-level to estimate the automaticity of skill control-xy-position.
2. Considering lesson hover to increase proficiency in skill hover.
3. Selecting best lesson: straight-level. Objective is to estimate-skill-automaticity skill control-xy-position. Importance = 3.

Lesson configuration - log of the AIS's reasoning

4. Proficiency in controlling altitude is high. Setting altitude tolerance to 20.0 feet.
5. Proficiency in controlling speed is high. Setting speed tolerance to 5.0 knots per hour.
6. Proficiency in controlling direction is high. Setting direction tolerance to 10.0 degrees.

Lesson briefing delivered to the student

Dave, please perform straight and level flight. Take off and reach altitude of 2000.0 plus or minus 20.0 feet. Maintain direction of 180.0 plus or minus 10.0 degrees. Accelerate to 50.0 plus or minus 5.0 knots per hour. During this task, please count from 1 to 1000 by 4.

Note

Program instructed the student to carry out an (admittedly unrealistic) auxilliary task of counting from 1 to 1000 by 4 to estimate the automaticity with which Dave can perform this task.

A.4 Student John

John is in his twentieth week of training. His performance today is poorer than usual.

Lesson selection - log of the AIS's reasoning

1. Proficiency of skill hover is medium and the nominal proficiency for students in week 20 of training is high.
2. Consider possibility that the cause of today's poor performance is that the student is tired.
3. Consider possibility that the cause of today's poor performance is low motivation.
4. Considering lesson hover to increase proficiency in skill hover.
5. Selecting best lesson: hover. Objective is to learn skill hover. Importance = 1.

Lesson configuration - log of the AIS's reasoning

6. Hover proficiency is medium. Setting altitude tolerance to 20.0 feet. Setting drift tolerance to 20.0 feet. Setting heading tolerance to 10.0 degrees.

Lesson briefing delivered to the student.

John, please perform a hover. Take off and reach altitude of 800.0 plus or minus 20.0 feet. Stay within 20.0 feet of the target hover position. Maintain heading of 180.0 plus or minus 10.0 degrees.

Note

Program displays a visual aid that shows a bird's eye view of the helicopter's position and heading relative to the hover target position and heading. The display shows standard drift tolerances as well as target tolerances based upon the student's current hover proficiency level.

A.5 Student Sue

Sue is in her tenth week of training. Her proficiency in hover flight is less than nominal.

Lesson selection - log of the AIS's reasoning

1. Her proficiency and automaticity in subskills of hover flight are high.
2. Proficiency of skill hover is low and the nominal proficiency for students in week 10 of training is medium.
3. Considering lesson hover to increase proficiency in skill hover.
4. Selecting best lesson: hover. Objective is to learn skill hover. Importance = 1.

Lesson configuration - log of the AIS's reasoning

5. Hover proficiency is low. Setting altitude tolerance to 100.0 feet. Setting drift tolerance to 100.0 feet. Setting heading tolerance to 20.0 degrees.

Lesson briefing delivered to the student.

Sue, please perform a hover. Take off and reach altitude of 800.0 plus or minus 100.0 feet. Stay within 100.0 feet of the target hover position. Maintain heading of 180.0 plus or minus 20.0 degrees.

Note

Program displays a visual aid that shows a bird's eye view of the helicopter's position and heading relative to the hover target position and heading. The display shows standard drift tolerances as well as target tolerances based upon the student's current hover proficiency level. Sue's proficiency is lower than John's, so the program selected wider target drift tolerances.